I/O NEWS

New Products
User Notes
New SUDS Releases

THE OFFICIAL PUBLICATION OF THE INTERNATIONAL ASSOCIATION OF CROMEMCO USERS

Volume Five, Number One

Single Copy Price $10.00

CE A New Text Editor for Cromemco's UNIX & Cromix-Plus Systems

The written word is the hallmark of man's ingenuity, and among the most powerful tool he has yet to devise. And because man spends so much time and expends so much energy with the written word, it comes as no surprise that one of his other super-tools — the computer — was applied to easing the burden of this relentless pursuit. Man had a better idea — the text editor — which Cromemco has continued to refine. The result is their new screen editor for UNIX and Cromix-Plus systems, CE.

Nearly every computer, from mainframe to micro, has some facility to enable a person to write text, store and retrieve it, make changes to it, format it and print it. They all differ in regards to ease of use, speed, flexibility and capability. What the designers at Cromemco have done with CE is to incorporate many of the power features of the cumbersome-to-use UNIX editors while retaining the many favorable features of their popular screen editor, and have tailored it to take advantage of todays intelligent terminals and their line of 68000-based supermicros. The design criteria that were met are ease of use, speed, power, and flexibility.

Relief for UNIX Users

Up to now, when you spoke of text editors for the UNIX System V operating system, you limited your conversation to ed and vi. Powerful and fast? Yes. Easy to use? Not hardly. Thats why you see all of the books about learning to use vi (or ed), and the coffee mugs decorated with vi commands, and that poster with vi commands spiraling within the chambers of a Nautilus shell, and the two-day seminars to get acquainted with the UNIX text editor, and ... (get the picture?). The most widely voiced complaint, and the greatest hindrance to the overall public acceptance of UNIX, is that it is so complex and difficult to learn and use. Both ed and vi contributed their fair share to this bad reputation.

On the other hand, when you speak of text editors for CDOS or Cromix you think of Screen. No two-day seminars required here. In fact, most people who use it have never even had to peek at the manual (maybe just a quick peek). The command structure and visual display are so straight forward that users can, with a minimum of experimentation, quickly become fully

Continued on page 28

CFSU
Cromix File Structure Utility for UNIX

by Rick Dhaenens

If you have upgraded your Cromemco computer to the Unix operating system — or have added a Unix system to an existing Cromix installation — then you probably have the same problem that I do. I have too much information to move!

I am a long time Cromemco user and have several megabytes of programs that I have developed over the years. Most of these programs (or the output of these programs) are useful on Unix as well as Cromix. The current method of moving information from Cromix to Unix is a little awkward. You have to use the TAR utility to copy the desired files to the disk common area while running Cromix, then boot to Unix where the files are brought back into the Unix file system. This has several disadvan-

Continued on page 8

Why VSL International Chose Cromemco

by Igor Uherkovich

VSL International is the head office of a world-wide network of companies operating in the construction industry with some special, complementary techniques. The most important of these techniques is the post-tensioning used, for example, in all concrete bridges and other sophisticated concrete structures. There are half a dozen internationally used prestressing systems: VSL is the leading company in this competition.

The market's focal points are the USA, Western Europe, South East Asia, Australia, and the Middle East. The annual turnover of the VSL-Group is nearly 200 million US dollars. VSL International is wholly owned by LOSINGER — the largest construction company of Switzerland — and LOSINGER itself is part of the ENSERCH CORPORATION,

Continued on page 13

ANNOUNCING!!

Powerful New Software Packages Including

RealWorld ™

on UNIX System V

RealWorld™

Software Standards announces their new line of RealWorld accounting packages for operation under Cromemco's new UNIX System V machines. These packages are written in powerful High-Performance Micro-Focus cobol and utilize their new super high-speed NATIVE-CODE GENERATOR. All packages are optimized for speed and support the full multi-user record level locking features of UNIX System V.

FEATURES:	PACKAGES INCLUDE:
Multi-User	General Ledger
Fully Integrated	Accounts Payable
Advanced Mini-Computer Design	Accounts Receivable
Password Protection	Payroll
Extensive Data File Integrity Check	Inventory Control
Backup/Restore Facilities	Order Entry / Billing
Native-Code Program Files	Sales Analysis

Software Standards also supports the RealWorld packages in full multi-user under 68000 Cromix as well as in single-user under Z80 Cromix

dB COMPILER

dBcompiler from WordTech Systems is a powerful dBase II compiler. Compiled dBase II application programs operate without the presence of dBase II. The compiler gives the programmer the ability to generate machine-efficient, stand alone, effectively 'encrypted' programs. Generally, applications will execute faster when compiled, and require less space. In addition, dBcompiler offers an amazing speed increase in both sort and indexing operations.

VE VT100 & VT52 emulators

Software Standards VT100 and VT52 terminal emulators give your Cromemco systems the power to communicate with large DEC minicomputers. Available for 68000 & Z80 CROMIX as well as the C-10.

Contact your Cromemco dealer for more information. If you don't have a Cromemco Dealer, contact Software Standards for the name of the dealer nearest you.

Software Standards, Inc.

6191 Choctaw • Baton Rouge, La. 70805
504/355-8024

UNIX is a registered trademark of AT&T.
DEC, VT100, and VT52 are registered trademarks of Digital Equipment Corp.
dBASE II is a registered trademark of Ashon-Tate.
Cromix and C-10 are trademarks of Cromemco, Inc.

RealWorld™ is a trademark of Realworld Corporation.
dB COMPILER is a trademark of WordTech Systems, Inc.

BEST IN SALES AND SERVICE

We understand your desire to have a local Cromemco dealer in your area, and we believe we have a perfect solution for your problem at this time.

We are THE MOST QUALIFIED facility in the U.S. as we are the first recipient of Cromemco's new award "Number 1 U.S. Dealer". Again we have received awards for "Outstanding Service" as well as ranking in the "Top 25 Dealers". Based in the greater Dallas area, we sell and service Cromemco equipment in all regions of the U.S., and in several foreign countries.

Due to our central location and the convenient access to DFW Airport, equipment can be received from any location in the U.S. and returned to the customer within three working days (excluding transit time), if our expedited service is requested.

Service and support are the reasons that we are successful at what we do. Simply said, sales backed by support means success and that means **Computer Crossroads**.

SPECIALS OF THE MONTH

MEMORY UPGRADE

512MSU / 2048 MSU.....CALL DANNEY FOR QUOTE
180 DAY WARRANTY ON OUR WORK AND THE INTEGRATED CIRCUITS THAT WE PUT ON YOUR BOARDS.

MODEMS

US ROBOTICS 2400/BELL 212A MODEM.....LIST 699.00.....**525.00**
HAYES SMARTMODEM COMPATIBLE.

HARD DISK

MAXTOR XT-1140 144 MB HARD DISK.....LIST 5,995.00.....**3,795.00**
USED BY CROMEMCO IN CS100, CS300 & CS400.

MICROPOLIS 1304 50 MB HARD DISK.....LIST 2,100.00.....**1,595.00**
USED BY CROMEMCO IN CS100, CS300 & CS400.

CHECK OUR SPECIAL DISCOUNT ON THE STDC CONTROLLER BOARD WHEN ORDERING ANY ST506 INTERFACE HARD DISK FROM US.

FLOPPY DISK

SHUGART SA455-3AA HALF HEIGHT 5¼".....LIST 250.00.....**150.00**
USED BY CROMEMCO IN CS100 & CS400.

TANDON TM100-2 FULL HEIGHT 5¼".....LIST 250.00.....**150.00**
USED BY CROMEMCO IN OLDER CS100 AND ALL CFD.

TANDON TM848-2E HALF HEIGHT 8".....LIST 695.00.....**450.00**
USE BY CROMEMCO IN CS3A, CS300 & CLD.

COMPUTER CROSSROADS of AMERICA, INC.

SOFTWARE
SERVICE
HARDWARE

Where It All Comes Together

We will meet or beat any bonafide sales quote on Cromemco equipment and will make available the best in support at no extra cost.

6 Terrace Shopping Center • Richardson, Texas 75081

FOR SUPER SERVICE CALL (214)231-6108 Telex: 4991118

table of contents

November/December, 1985 Volume Five, Number One

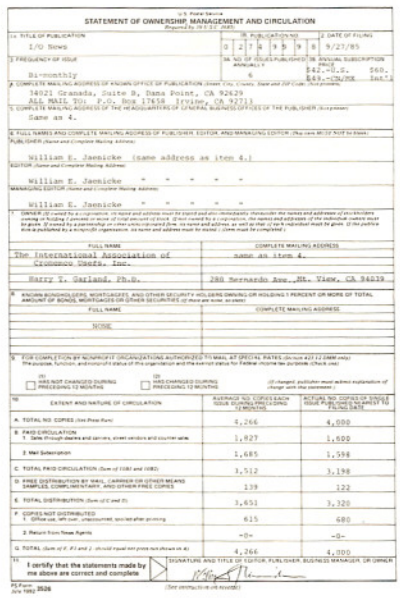
I/O NEWS

The Official Publication of The International Association of Cromemco Users is available through membership in the association. Editorial and advertising policies are designed for the enlightenment of the members in regard to new uses for, and developments of, Cromemco products and other products compatible with Cromemco systems.

COVER FEATURES

ARTICLES & FEATURES

DEPARTMENTS



STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

1A. TITLE OF PUBLICATION	1B. PUBLICATION NO. 0 2 7 4 9 9 8
I/O News	2. DATE OF FILING 9/27/85
3. FREQUENCY OF ISSUE: Bi-monthly	3A. NO. OF ISSUES PUBLISHED ANNUALLY: 6
	3B. ANNUAL SUBSCRIPTION PRICE: $42.-U.S. $60. $49.-CN/MX Intl.

4. COMPLETE MAILING ADDRESS OF KNOWN OFFICE OF PUBLICATION (Street, City, County, State and ZIP Code) (Not printers): 34021 Granada, Suite B, Dana Point, CA 92629
ALL MAIL TO: P.O. Box 17658 Irvine, CA 92713

5. COMPLETE MAILING ADDRESS OF THE HEADQUARTERS OR GENERAL BUSINESS OFFICES OF THE PUBLISHER (Not printer): Same as 4.

6. FULL NAMES AND COMPLETE MAILING ADDRESS OF PUBLISHER, EDITOR, AND MANAGING EDITOR (This item MUST NOT be blank)

PUBLISHER (Name and Complete Mailing Address): William E. Jaenicke (same address as item 4.)

EDITOR (Name and Complete Mailing Address): William E. Jaenicke " " " "

MANAGING EDITOR (Name and Complete Mailing Address): William E. Jaenicke " " " "

7. OWNER (If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be given. If the publication is published by a nonprofit organization, its name and address must be stated.) (Item must be completed.)

FULL NAME	COMPLETE MAILING ADDRESS
The International Association of Cromemco Users, Inc.	same as item 4.
Harry T. Garland, Ph.D.	280 Bernardo Ave., Mt. View, CA 94039

8. KNOWN BONDHOLDERS, MORTGAGEES, AND OTHER SECURITY HOLDERS OWNING OR HOLDING 1 PERCENT OR MORE OF TOTAL AMOUNT OF BONDS, MORTGAGES OR OTHER SECURITIES (If there are none, so state)

FULL NAME	COMPLETE MAILING ADDRESS
NONE	

9. FOR COMPLETION BY NONPROFIT ORGANIZATIONS AUTHORIZED TO MAIL AT SPECIAL RATES (Section 422.12 DMM only)
The purpose, function, and nonprofit status of this organization and the exempt status for Federal income tax purposes: (Check one)
☐ (1) HAS NOT CHANGED DURING PRECEDING 12 MONTHS ☐ (2) HAS CHANGED DURING PRECEDING 12 MONTHS (If changed, publisher must submit explanation of change with this statement.)

10. EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES (Net Press Run)	4,266	4,000
B. PAID CIRCULATION 1. Sales through dealers and carriers, street vendors and counter sales	1,827	1,600
2. Mail Subscription	1,685	1,598
C. TOTAL PAID CIRCULATION (Sum of 10B1 and 10B2)	3,512	3,198
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	139	122
E. TOTAL DISTRIBUTION (Sum of C and D)	3,651	3,320
F. COPIES NOT DISTRIBUTED 1. Office use, left over, unaccounted, spoiled after printing	615	680
2. Return from News Agents	-0-	-0-
G. TOTAL (Sum of E, F1 and 2 - should equal net press run shown in A)	4,266	4,000

11. I certify that the statements made by me above are correct and complete. SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER: [signature]

PS Form 3526 (See instructions on reverse)

I/O NEWS [ISSN 0274-9998] is published bi-monthly by The International Association of Cromemco Users (a California corporation), P.O. Box 17658, Irvine, CA 92713. General Offices are at 34021 Granada #B, Dana Point, CA 92629. Telephone: (714) 661-9764. Second Class Postage Paid at Dana Point and Santa Ana, CA.
POSTMASTER: Send address changes to I/O NEWS, P.O. Box 17658, Irvine, CA 92713.

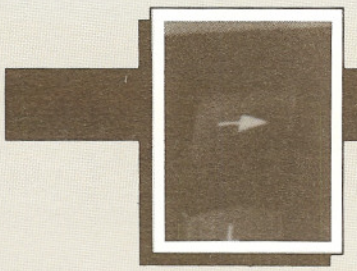
Subscriptions to I/O NEWS are entered with membership in The IACU. Yearly memberships may be purchased for $42 (U.S. delivery address), $49 (delivery address in Canada or Mexico), and $60 (other international delivery addresses). Contact IACU for multi-year membership rates. Back issues of I/O NEWS are available for $10.00 per issue. Please note: all prices are in U.S. dollars.

Return postage must be included with all manuscripts and photos submitted if they are to be returned. The IACU and I/O NEWS accept no responsibility for the return of unsolicited materials.

William E. Jaenicke
Editor and Publisher

Lisa B. Jaenicke
Business Manager

Art Direction: Howard Millman
Printing: Hallmark Litho
Typography: Western Outdoors



INPUT...

Editor:

I use a Cromemco Z2D with two 5¼ " floppy disks and a 5 megabyte hard disk. Perhaps you can help me clarify a dilemma about utilizing programs for IBM-PC computers. One solution that came to mind was to purchase an IBM-PC andinteract with my Cromemco files through a modem. Most of these are in dBASE II. A second option, about which I only recently learned, is to obtain an S-100 IBM-PC board to which can be connected an IBM compatible keyboard and monitor. These S-100 boards are made by Lomas Data Products, Inc., 182 Cedar Hill Road, Marlborough, MA 01752.

I have read over some of their technical information but I am unclear about how to coordinate the two boards on one S-100 bus. Are you aware of any S-100 dual processors containing the Z80-A and an 8086 or 80286 processor so that I could run both my CP/M programs at the same time as the IBM-PC programs. This would require a concurrent-dos operating system and would be required to drive my two floppy disks and my hard disk.

It seems to me that this conversion must have been thought about or done by others. Unfortunately for me, Cromemco only makes a Z80/68000 coprocessor and I doubt that they plan to change to meet my needs. Any information, advice or assistance you or other Cromemco users could give would be most appreciated.

Cordially,

Herbert Selenkow, M.D.
BRIGHAM MEDICAL GROUP
454 Brookline Ave.
Boston, MA 02215
(617) 732-5500

The following letter was originally submitted to S-100 Journal. *A copy was forwarded to* I/O News *by its' author (and I.A.C.U. member), Ben Ross Schneider.*

Ed.

Mr. Jay Vilhena, Editor
S-100 Journal
2426 Wade Avenue
Raleigh, NC 27607

Dear Mr. Vilhena:

As Emerson said to Whitman "I greet you at the beginning of an auspicious career!" The first issue of *S-100 Journal* looks beautiful, and it reads very well. A veteran English teacher doesn't often say that about computer journals. But most of all I admire your fighting spirit. IEEE-696 provides the only truly viable base for the future development of computing. The "IBM standard," continually undercut by IBM's attempts to get a proprietary hold, never will be the means of liberating the creative energies of our computing community. Countless man-hours have been wasted to get around its hidden quirks, and only the most advanced engineers with millions to waste can play this game. But despite S-100's obvious superiority, I believe it is doomed unless those who believe in it rise to the occasion.

I have been hacking about in computerland since 1965. As you will see if you read my book, *My Personal Computer*, I like to work things out on my own terms. Now the principal makers of S-100 boards — Cromemco, CompuPro, Lomas, MacroTech, and their dealers — are telling me that I can't make my S-100 system IBM-compatible for less than it takes to buy an off-the-shelf clone! Their literature is cryptic and incomplete. Their tech support people can't answer my questions. This state of affairs is unbelievable, suicidal, intolerable, and immoral. IBM has virtually ruined the market for state-of-the-art Z80 software. Now it endangers the S-100 bus itself. Apple has been badly hurt by Steve Jobs' insistence that the Mac be a closed box, and IBM is tightening its stranglehold. By controlling the kind of hardware and software that evolve, they control what kind of information is processed. That's why inertia in the S-100 community may bring the end of liberty in computing.

The only way for S-100 to beat IBM is to join the crowd, and help the clone makers take the IBM standard away from IBM. The user who wants state-of-the-art software absolutely must become IBM-compatible. Ideally, he needs a box in which he can mount any available piece of software for any processor at a lower price than it would take to buy a new computer. Look at the HiFi business. Early in the game the closed box gave way to separate components. Americans love to do it themselves. They refuese to be forcefed. S-100 is the American way. But S-100 suppliers will lose the fight unless they wake up to the fact that the mass market is there for the taking. Their best weapons are better boards, better documentation, lower prices, and mass advertising. Do they have the guts, or will they take the money and run?

Yours sincerely,

Ben Ross Schneider, Jr
Department of English
Lawrence University
Box 599
Appleton, WI 54912

The preceding letters are representative of a growing concern about IBM compatibility on S-100 based computers in general, and Cromemco systems in particular. I/O NEWS *intends to explore the various methods available for achieving this compatibility in this issue, and in future issues.*

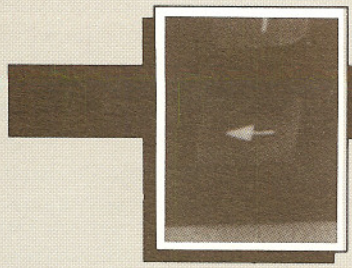
At present there appear to be three distinct approaches. The first utilizes specialized communciations software which enables the PC to act as a workstation on a larger Cromemco system. This is exemplified by software such as PC-Works, Tango, and ProCall-PC. In this configuration the MS-DOS applications are run under the PC directly: the Cromemco system is employed as a file and resource handler.

The second alternative requires additional hardware in the form of S-100 processor boards based on the Intel 8086 and 80286 microprocessors. In this environment, MS-DOS would be run directly, while CP/M could be emulated. We have learned of a manufacturer of such a board, in addition to the one mentioned by Dr. Selenkow, and are awaiting a more detailed product description.

The third avenue to compatibility arises by virtue of application software that has been ported to run under MS-DOS and other operating systems such as CP/M, CDOS, Cromix, and UNIX. In the last issue we announced such a product — the Cromix/CDOS/MS-DOS version of 32K Structured BASIC. Applications programs such as dBASE III, Informix, and TODAY will have MS-DOS, UNIX, and hopefully, Cromix counterparts. Others may follow suit.

If you know of other alternatives, let us know. We will continue to pursue our investigations and report our findings in upcoming issues.

Ed.



OUTPUT...

The publication of this issue, Volume V, Number 1, for November/December 1985, marks the beginning of the fifth publication year for I/O NEWS. So, as a birthday present of sorts, we gave the magazine a new look. Out with the old, and in with the new.

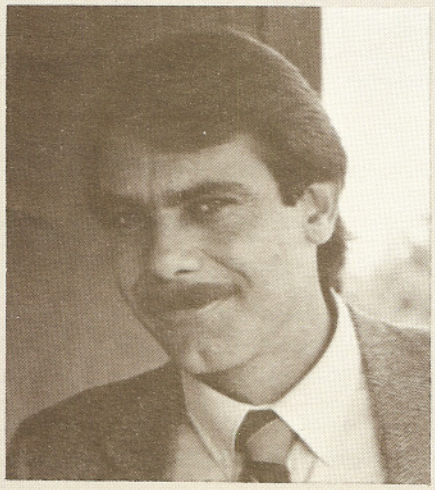
At first glance, your eyes may deceive you — you may ask yourself, "Does that read 'I/O NEXUS'?" That's what I saw when I looked upon the new logo for the first time. In the end, that little trick of the eye is what sold me on the design. It prompted me to reach for my handy Funk & Wagnalls and confirm the definition I so vaguely recalled.

nex•us (neksəs) *n. pl.* **•us•es** or **•us** A bond or tie between the several members of a group or series; link [L *nectare* tie]

I was taken by how succinctly the definition of 'nexus' described the purpose of The I.A.C.U. and I/O NEWS. A bond or tie between Cromemco Users, Cromemco Dealers, and Cromemco. An information link — a news link — an experience link. I/O NEWS/NEXUS. What could be more appropriate?

One aspect of the nexus involves getting first-hand information directly from the source — Cromemco. Of course, the best way to do this is to drop in every once in a while. Lisa and I had the pleasure of visiting Cromemco recently, which resulted in some of the information presented in this issue. During the visit, we had the opportunity to meet with Cromemco's President, Dr. Harry Garland, and received a most interesting tour of the manufacturing facility. We saw the elaborate mechanisms and procedures for quality control that have resulted in Cromemco's reputation for quality and reliability.

We also sat in on a meeting between the various regional office technical managers and the in-house technial support and project managers. It is in meetings such as this that customer feedback is digested. The level of technical expertise brought together was immense. Problems detected in the field were voiced, and in many cases, solutions were immediately devised. The more involved problems were prioritized; an overall game plan was developed.



Bill Jaenicke

Lisa Jaenicke

It was encouraging to see this type of interaction, to witness so much experience and know-how applied towards the improvement of quality. It became immediately apparent that Cromemco truely does care about customer satisfaction, that they are responsive to complaints arising in the field. They really do try harder. Excellence in microcomputers doesn't come easy.

What's happening at Cromemco, while interesting and important, is only a part of the information flowing through the nexus. The greater proportion is generated by you — the ones actually using the systems. Much of this finds its way into I/O NEWS, either as an article, a contribution to a regular column, or mention in Bits & Bytes. This is where I/O NEWS shines.

With this issue we have two new stars in our constellation. Jordan Siedband, a long-time member and past contributor, has taken the helm of the regularly appearing INSIDE CROMIX column. Jordan's experience with the evolution of Cromix will be a great asset. I think all Cromix users will appreciate his first contribution — a UNIX-like "calendar" utility.

As UNIX is seeing more and more use among Cromemco users, I'm happy to announce that we will soon have a column devoted to it. Rick Dhaenens, Technical Support Manager at the Cromemco Regional Office in Atlanta, Georgia, will be its editor. For this issue, Rick has contributed his Cromix File Structure Utility (CFSU), which appears on the front cover. I/O NEWS is extremely fortunate and deeply grateful that Rick has volunteered his services — as far as UNIX on Cromemco machines is concerned, there's nobody more knowledgeable.

Elsewhere in this issue you'll learn about Cromemco's new UNIX/Cromix text editor, CE. It combines the power of vi with the ease of use of screen. There's also news about upcoming SUDS updates, a printer conversion program for WriteMaster, and lots on new products. We hope that you like the content — and the new container!

With best wishes for the New Year,

William E. Jaenicke
Editor & Publisher

Funk & Wagnalls
New Comprehensive International Dictionary of the English Language

New SUDS Releases
68000 Software Greatly Enhanced

A large number of software updates are being sent to appropriate SUDS subscribers. Because of the large number, and since many are new versions of the Cromemco 68000 languages, the following summary of these latest updates is provided. Detailed information for each package is contained in the appropriate Software Update Service Notes provided with the SUDS update.

Cromemco UNIX Operating System (Model UNIX-X-S/L)

Release 3 of the Cromemco UNIX Operating System is now available and provides the following new and revised features:

1. New and improved handling of all serial I/O. Throughout for the serial channels has been significantly increased (up to 8-fold). UNIX tty capabilities have been fully implemented, including modem support, parity selection, etc. Baud rates of up to 19200 are selectable on the Octart. The console port now supports C-5/C-10 function keys.
2. Nine track tape is now supported under UNIX.
3. The Cromemco Cartridge Tape Drive (CTD) is now fully supported under UNIX.
4. In conjunction with RDOS 3.12, it is now possible to boot both UNIX and Cromix-Plus directly to STDC hard disk partitions. RDOS 3.12 is available for purchase from Cromemco.
5. I/O processor downloading under UNIX. There are currently 3 types of input/output processor boards supported under UNIX: **Octart** (serial lines ie. printers, modems, terminals), **IOP/CSP** (9 track tape) and **Biart** (3270 bisync and X.25 communications only). The programs which execute on these boards can now be downloaded directly from UNIX.
6. Error correcting memory (ECC) can now be controlled and monitored under UNIX using the ecc program.
7. Support for Cromemco 3270 bisync and X.25 communication software has been added to the UNIX kernel. These packages are available for purchase separately from Cromemco (model numbers: 3270BSC-XL/S and X.25-XL/S).
8. The number of files that a process may have open at one time has been increased from 20 to 32.
9. General periodic upgrade and maintenance has been performed on the standard UNIX System V utilities.

68000 Pascal Compiler (Model PAS-D-S/L)

Release 6 (version 2.42) of the 68000 Pascal programming language has few changes from the user's persepctive, but provides substantial improvements in speed and efficiency.

68000 Fast C Compiler (Model FSTCCC-D-S/L)

Release 3 (version 2.41) of the 68000 Fast C programming language supports Cromemco's Maximizer co-processor board. The speed and efficiency of the Fast C compiler has been substantially improved, and the Symbolic Debugger has been slightly modified.

68000 Fast FORTRAN Compiler (Model FSTFOR-D-S/L)

Release 4 (version 2.41) of the 68000 Fast FORTRANN-77 programming language provides a substantial increase in speed and efficiency and a variety of new and revised features. Source code compiled by earlier versions of Fast FORTRAN-77 is incompatible with version 2.41, and should be recompiled.

68000 Fast Pascal Compiler (Model FSTPAS-D-S/L)

Release 4 (version 2.42) of the 68000 Fast Pascal programming language has few changes from the user's persepctive, but provides substantial improvements in speed and efficiency.

68000 Fast BASIC-Plus Interpreter (Model FSTBAS-D-S/L)

Release 2 (version 2.40) of the 68000 Fast BASIC-Plus programming language supports Cromemco's Maximizer co-processor board. The speed and efficiency of the interpreter has been improved.

Cromix-Plus Operating System (Model CRO-PLUS-XCL/S)

Release 5 (version 31.04) of the Cromix-Plus Operating System is now available and provides the following new and revised features.

1. New utilities and Shell commands: ce.bin (see front cover); convert.bin — performs variety of file transformations (replaces the Ddump utility); diskinfo.bin — displays the parameters, partition table, and alternate track table of an initialized hard disk; ftar.bin — creates and restores file archives on any file or device (primarily intended for cartridge tape drives); inithard.bin — initializes SMD, WDI-II, and STDC hard disk drives (replaces initstdc utility); term.bin — displays or changes the name of the user's terminal. Part of the new UNIX-like termcap feature of Cromix-Plus.
2. Revised utilities: ccall.bin, help.bin, inittape.bin, input.bin, mode.bin, patch.bin, priority.bin, rcopy.bin, readall.bin, shell.bin, and sort.bin.
3. New files: the new /etc/termcaps file describes terminal capabilities, and the new uboot.sys program boots the UNIX System V Operating System.
4. New system calls: _ptrace, _uchstat, and _ustat (the Z80 and 68000 equate files have been updated for new modes and system calls).
5. New device driver: the hd driver allows the use of IMI 5, 10, or 20 Mbyte hard disks. This driver requires WDI-II and DPU boards; it is not compatible with WDI (original) or XPU hardware.
6. Direct booting of Cromix-Plus to STDC hard disks, in conjunction with the new release of RDOS version 3.12, is now supported.

68000 C Compiler (Model CCC-D-S/L)

Release 4 (version 2.41) of the 68000 C programming language is now available. Though there are few changes from the user's perspective, the speed and efficiency of the C compiler has been substantially improved, and the Symbolic Debugger has been slightly modified.

C-Isam File Access (Model CISAM-D-S/L)

This new release (version 2.04) makes C-ISAM-D compatible with INFORMIX version 3.3.

C-Isam File Access (Model CISAM-S-X/L)

This new release (version 2.04) makes C-ISAM-X compatible with INFORMIX version 3.3.

68000 Basic Interpreter (Model BAS-D-S/L)

Release 2 (version 2.40) of the 68000 BASIC-Plus programming language is now available; this new BASIC interpreter provides substantial improvements in speed and efficiency.

68000 FORTRAN 77 Compiler (Model FOR-D-S/L)

Release 7 (version 2.41) of the 68000 FORTRAN 77 programming language provides a substantial increase in speed and efficiency, as well as a variety of new and revised features. Any source code compiled by earlier versions of FORTRAN 77 is incompatible with ver-

Continued on page 32

UNIX CFSU

Continued from front cover

tages: first it is very time consuming and you always forget to bring the one file that you need; second, the files that I want to move always seem to be about 20% bigger than the available common space.

As the Southern Regional Technical Support Manager for Cromemco I talk to many of our O.E.M.'s and dealer's technical people. Many useful utilities have been developed by these people; one of these is the Cromix File Structure Utility (CFSU) developed by Chip Salzenberg for Alphatype Corp.

This utility was written to allow the C-10 to read a diskette with a Cromix file system. The program was originally written with a Z-80 'C' compiler and and linked to an assembly language routine that talked directly to the WD-1793 disk controller chip in the C-10. The user interface was written to emulate the Cromix shell and several of the Cromix utilities. There are several operational differences but the basic commands dir, ls, and type work pretty much as expected. The copy command copies files from the Cromix file system to the Unix file system. Help is available with the help command and exit will take you back to Unix.

This was just what I needed! Rather than continue to inflict pain and agony on myself every time I needed another file from Cromix I decided to port CFSU to Unix to allow access to the Cromix partition of the hard disk. I would have liked to allow access to floppies also but the Unix floppy drivers do not know about the Cflop single density track 0.

There were several things that needed to be changed for CFSU to be used on Unix. The first thing was the routine to read a block from the floppy. This routine was replaced by a command line option to open a Unix block device that has a Cromix file system on it. If no arguments are used the default device used is /dev/cromix which may be linked to the appropriate device. I also wanted to make the display format of the directory listings look like the Cromix ls -m command. I also made several other modifications and added some comments to the program.

The source code for CFSU is split into 2 files: CFSU.H which is the definition of the Cromix superblock, inode, and directory structures, and CFSU.C which is the 'C' source code for the utility. Since CFSU is based on the Cromix file structure a review is in order.

The Cromix file system is based on a disk block size of 512 bytes. Block devices are accessed starting with block 0 up to the limit of the device size. Inodes are 128 bytes long so there are 4 inodes per block. Directory entries are 32 bytes long for a total of 16 directory entries per block. These and other definitions are available in the Cromix include file STRUCTS.H. Information from this file was used to make the CFSU.H include file.

The superblock is a description of the file system and is located on block 1 of a device. It contains values for the total number of blocks for this file system, the number of inodes, the starting block of the inodes, and the free list pointers for the block and inode free lists.

The first directory is the root directory and is referenced by the first inode. This inode's block pointers point to the blocks used to hold the directory entries for the files in the root directory. Each directory entry has 24 bytes for the name of the file and a pointer to the inode used for the file.

Inodes are used to represent the three basic types of Cromix entities *files*, *directories*, and *devices*. There is one inode for each entity, and there are minor differences in how they are used for each type. The inode contains general information such as the size of the file, the creation, modification, and dump times, and the block pointers to the actual file data.

File data is accessed by using the *block pointers* in the inode. Each inode only has 20 block pointers but by using levels of indirection, pointers that point to blocks of pointers, those 20 pointers can describe incredibly large files. This method of file access is used by both Unix and Cromix.

The first 16 blocks (8,192 bytes) of the file, the inode block pointer points directly to file data. The 17th inode block pointer has 2 levels of indirection: it points to a block that is used as pointers to the next 128 blocks (65,535 bytes) of file data. The 18th inode block pointer has 3 levels of indirection: it points to a block of pointers that is used to point to 128 blocks of pointers that each point to 128 blocks of file data, this corresponds to 16,384 blocks of storage (8,388,608 bytes). CFSU only implements 3 levels of indirection which limits the file sizes that may be accessed to 8,462,335 bytes. Cromix implements 4 levels of indirection with the 19th pointer for a file size of over 1 gigabyte.

To compile CFSU on Unix requires the Programmers Software Tools (PST). Type cc -O -s -o cfsu cfsu.c to compile and generate the CFSU executable file. To execute type cfsu devname.

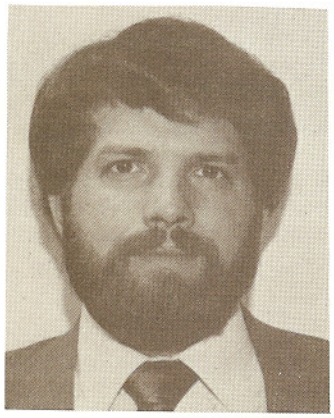
CFSU has been a big help in the porting of software from Cromix to Unix and is useful anytime both environments are used on the same machine. It is also interesting to note that CFSU can be compiled and run with no changes by the SVS 'C' compiler running under Cromix. The complimentary program UFSU to allow access to Unix files on Cromix is an easy task using these same concepts.

```
/*************************************************************************/
/*                    Cromix file system utility.                        */
/*            Written by: Chip Salzenberg for Alphatype Inc.             */
/*                 Modifications by: Rick Dhaenens                       */
/*                                                                       */
/*         This program may be used for non-commercial purposes.         */
/*************************************************************************/

#include <stdio.h>                      /* Include standard I/O routines.   */
#include "cfsu.h"                       /* Include Cromix file structs.     */

int ver = 1;                            /* This program's version number.   */
int rev = 0;                            /* This Program's revision number.  */

#define CMDMAX  80                      /* Maximum command line length.     */
#define ARGMAX  5                       /* Max number of command line args. */
#define CMDS    15                      /* Number of commands in list.      */

int fd;                                 /* Global Cromix device discriptor. */
int curdir = 1;                         /* Global inode number of cur dir.  */
int fs_isize, fs_fsize, fs_istrt;       /* Global disk parameters.          */

int no_blocks();                        /* Forward declarations.            */


main(argc, argv)                        /* Main command dispatch loop.      */
int argc;
char *argv[];
{
  int cmd_argc, i;                      /* Define main variables.           */
  SUPERBLK superblk;
  char cmd_line[CMDMAX], *cmd_argv[ARGMAX];
  int ls(), cd(), cat(), copy(), dd(), help(), pexit();

  static struct {                       /* Define command list structure.   */
    char *name;                         /* Array of routine name pointers   */
    int (*fn) ();                       /* and pointers to the function.    */
  } commands[] = {
    "l", ls, "ls", ls,                  /* Initialize commands in list.     */
    "d", cd, "cd", cd,                  /* Note: some are aliases.          */
    "ty", cat, "type", cat, "cat", cat,
    "cp", copy, "copy", copy,
    "dd",dd, "ddump", dd,
    "ex", pexit, "exit", pexit,
    "h", help, "help", help
  };
                                        /* Print signon message.            */
  printf("\n Cromix File System Utility.  Version %2d.%02d\n", ver, rev);

  if (argc != 2) {                      /* Open Cromix file structure dev.  */
    if ((fd = open("/dev/cromix", 0)) == ERR) {
      fprintf(stderr, "\n CFSU: Can't open '/dev/cromix'.\n\n");
      exit(ERR);
    } else
      printf("\n The Cromix file system device is '/dev/cromix'.\n\n");
  } else {
    if ((fd = open(argv[1], 0)) == ERR) {
      fprintf(stderr, "\n CFSU: Can't open '%s'.\n\n", argv[1]);
      exit(ERR);
    } else
      printf("\n The Cromix file system device is '%s'.\n\n", argv[1]);
  }

  readblk(&superblk, 1);                /* Read superblock from the device. */

  fs_istrt = superblk.s_istart;         /* Get start of inodes on disk.     */
  fs_isize = superblk.s_isize;          /* Get number of inodes on disk.    */
  fs_fsize = superblk.s_fsize;          /* Get number of blocks on disk.    */

                                        /* Print disk statistics.           */
  printf(" Inodes start on block %d.\n", fs_istrt);
  printf(" Total disk size = %d bytes.\n\n", 512*(fs_fsize-(fs_isize/4)));
  printf(" Maximum inode number = %d.\n", fs_isize);
  printf(" Maximum block number = %d.\n\n", fs_fsize);

  /* This is the main loop to get and execute commands.                    */

  while ((printf("-> "), gets(cmd_line, CMDMAX)) != NULL) {

    parse(cmd_line, &cmd_argc, cmd_argv, ARGMAX);  /* Parse commands.    */

    if (cmd_argc == 0)                  /* If null command ...              */
      continue;                         /* Get another command.             */

    for (i = 0; i < CMDS; ++i) {        /* Go through the command list.     */

      if (strcmp(cmd_argv[0], commands[i].name) == 0) {  /* Test match.  */
          (*commands[i].fn) (cmd_argc, cmd_argv);  /* Yes execute it.      */
          break;
      }
    }

    if (i == CMDS)                      /* We made it with no match so:.    */
      fprintf(stderr,"\n CFSU: Unknown command '%s'.\n", cmd_argv[0]);

  }                                     /* If we're here there is trouble.  */
  fprintf(stderr,"\n CFSU: Command input error.\n\n.");
  close(fd);
  exit(ERR);
}


parse(buffer, cnt, parray, max_arg) /* Parse the command line with Unix */
int *cnt, max_arg;                      /* style argc and *argv[].          */
char *buffer, *parray[];
{
  char *cpos = buffer;                  /* Define local variables.          */

  *cnt = 0;                             /* Start with arguement 0.          */
  while (*cpos && *cnt < max_arg) {     /* Loop until null or max arg cnt.  */

    while (isspace(*cpos))              /* Wait for beginning of word.      */
      ++cpos;

    if (*cpos == '\0')                  /* Check for end of string.         */
      break;

    parray[(*cnt)++] = cpos;            /* Copy pointer to arg into arrray. */

    while (*cpos && ! isspace(*cpos))
      ++cpos;                           /* Read to the end of a word.       */

    if (*cpos == '\0')                  /* Check for the end again.         */
      break;

    *cpos++ = '\0';                     /* Add null terminator to the end.  */
  }

pexit()                                 /* Make a clean exit back to Unix.  */
{
  close(fd);                            /* Close the Cromix device.         */
  printf("\n");                         /* Make it look good.               */
  exit(0);                              /* Return to Unix.                  */
}


cd(cc, cv)                              /* Change directory command.        */
int cc;
char *cv[];
{
  int i;                                /* Define local variables.          */

  if (cc != 2) {                        /* Check argc for this command.     */
    printf(" DIR: Wrong number of arguements.\n");
    printf(" Syntax: DIR <pathname>.\n");
    printf("         CD  <pathname>.\n");
    printf("         D   <pathname>.\n");
    return;
  }

  if ((i = find_in(cv[1], 1)) == 0) /* Find inode for new directory.      */
    fprintf(stderr, " CFSU: Directory not found '%s'.\n", cv[1]);
  else
    curdir = i;                         /* Set current directory.           */
  return;
}


cat(cc, cv)                             /* Type a file to the screen.       */
int cc;
char *cv[];
{
  BLOCK buffer;                         /* Define local variables.          */
  INODE filenode;
  int i, thissize;
  int file_in, lastsize, arg;
  int fsize, st_blk, blk, blocks;

  if (cc < 2) {                         /* Check argc for this command.     */
    printf(" TYPE: Wrong number of arguements.\n");
    printf(" Syntax: TYPE <pathname> ... [pathname].\n");
    printf("         CAT  <pathname> ... [pathname].\n");
    printf("         TY   <pathname> ... [pathname].\n");
    return;
  }

  file_in = find_in(cv[1], 0);          /* Find inode for this file.        */

  if (file_in == 0) {                   /* Check for errors.                */
    fprintf(stderr, " TYPE: File not found '%s'.\n", cv[1]);
    return;
  }

  read_in(&filenode, file_in);          /* Read inode for this file.        */

  if ((filenode.i_stat & is_type) != is_ordin) {
    fprintf(stderr, " TYPE: Not ordinary file '%s'.\n", cv[arg]);
    return;                             /* Print error if not plain file.   */
  }

  fsize = filenode.i_size;              /* Get total file size.             */
  blocks = no_blocks(fsize);            /* Translate that into blocks.      */

  if ((lastsize = (fsize & 511)) == 0)  /* Get size of last block.          */
    lastsize = 512;                     /* Last block is 512 if even.       */

  for (blk = 0; blk < blocks; ++blk) {  /* Loop for all blocks.             */

    readiblk(buffer, &filenode, blk);   /* Read indirect file block.        */
    thissize = (blk == (blocks - 1)) ? lastsize : 512;

    for (i=0; i < thissize; i++)        /* Loop for characters in block.    */
      putchar(buffer[i] & 0x7F);        /* Write characters to screen.      */
  }
}


ls(cc, cv)                              /* List directory contents.         */
int cc;
char *cv[];
{
  int d, ls_in;                         /* Define local variables.          */
  int b, blocks;
  INODE dirnode;
  INODE filnode;
  DIRENTRY dirblock[NUMDIRS];

  if (cc == 1)                          /* If no args list current dir.     */
    ls_in = curdir;
  else {                                /* Else find inode and list dir.    */
    if ((ls_in = find_in(cv[1], 1)) == 0) {
      fprintf(stderr, " LS: Directory not found '%s'.\n", cv[1]);
      return;
    }
  }

  read_in(&dirnode, ls_in);             /* Read the directory inode.        */
  blocks = no_blocks(dirnode.i_size);   /* Find out how many blocks.        */

  for (b = 0; b < blocks; ++b) {        /* Loop for all blocks.             */
    readiblk(dirblock, &dirnode, b);    /* Read directory block.            */

    for (d = 0; d < NUMDIRS; ++d) {     /* Loop for # of dirs in block.     */

      if (dirblock[d].d_stat & ds_alloc) {   /* Check if allocated.         */
        read_in(&filnode, dirblock[d].d_inode);  /* Read file inode.       */

        switch (filnode.i_stat & is_type) {  /* Switch on file type.       */

          case is_direct:               /* Ls format for directory.         */
            printf("%lld D %2d ",filnode.i_dcount,filnode.i_nlinks);
            break;
          case is_ordin:                /* Ls format for ordinary file.     */
            printf("%llld   %2d ",filnode.i_size,filnode.i_nlinks);
            break;
          case is_block:                /* Ls format for block device.      */
            printf("    %3d:%-3d B %2d ",(filnode.i_dcount&0xff00)>>8,
                  (filnode.i_dcount&0xff),filnode.i_nlinks);
            break;
          case is_char:                 /* Ls format for char device.       */
            printf("    %3d:%-3d C %2d ",(filnode.i_dcount&0xff00)>>8,
                  (filnode.i_dcount&0xff),filnode.i_nlinks);
            break;
          default:                      /* No match so keep format straight */
            printf("    Unknown type ");
            break;
        }
        printf("%s\n",dirblock[d].d_name);  /* Print the filename.          */
```

```
    }
  }
}


copy(cc, cv)                            /* Copy Cromix file to Unix disk.   */
int cc;
char *cv[];
{
  BLOCK buffer;                         /* Define local variables.          */
  INODE filenode;
  char *destname;
  int i, thissize;
  int file_in, lastsize, dest_fd;
  int fsize, blk, st_blk, end_blk, blocks;

  if (cc < 3) {                         /* Check argc for this command.     */
    printf("COPY: wrong number of arguements.\n");
    printf("Syntax: COPY <Cromix filename> [Unix filename].\n");
    printf("        CP <Cromix filename> [Unix filename].\n");
    return;
  }

  if ((file_in = find_in(cv[1], 0)) == 0) {  /* Find inode for file.    */
    fprintf(stderr, " CFSU: File not found '%s'.\n", cv[1]);
    return;                             /* Print error if not found.        */
  }

  if ((dest_fd = creat(cv[2])) == ERR) {  /* Create Unix file.           */
    fprintf(stderr," CFSU: Error creating Unix file '%s'.\n", destname);
    return;                             /* Return on any create errors.     */
  }

  read_in(&filenode, file_in);          /* Read inode for file.             */
  fsize = filenode.i_size;              /* Get total file size.             */
  blocks = no_blocks(fsize);            /* Find the number of blocks.       */

  if ((lastsize = (fsize & 511)) == 0)  /* Get size of last block.          */

    lastsize = 512;

  for (blk = 0; blk < blocks; ++blk) {  /* Loop from beginning to end.      */

    readiblk(buffer, &filenode, blk);   /* Read indirect file block.        */
    thissize = (blk == blocks - 1) ? lastsize : 512;

    if ((i = write(dest_fd, buffer, thissize)) != thissize || i == ERR) {
      fprintf(stderr," COPY: Unix write error %d transfer aborted.\n", i);
      break;                            /* Print error message. Then quit.  */
    }
  }
  close(dest_fd);                       /* Close the Unix destination file. */
  return;
}


dd(cc, cv)                              /* Routine for direct block dump.   */
int cc;
char *cv[];
{
  int block;
  BLOCK buffer;

  if (cc != 2) {                        /* Check the arg count for command. */
    printf(" DDUMP: wrong number of arguements.\n");
    printf(" Syntax: DDUMP <disk block>.\n", fs_fsize);
    return;
  }

  sscanf(cv[1], "%d", &block);          /* Get the disk block number.       */

  if ((block < 0) || (block > fs_fsize)) {  /* Check for valid block #.     */
    fprintf(stderr," CFSU: Invalid block number %s\n", block);
    printf("       Block numbers range from 0 to %d.\n", fs_fsize);
    return;                             /* If problems occur tell about it. */
  }
                                        /* Print header then dump the block.*/
  printf("\nBlock %d byte %xh %d.\n\n", block, block * 512, block * 512);
  readblk(buffer, block);               /* Read block from disk.            */
  dumpblk(buffer, BLOCKSIZ, (block * BLOCKSIZ));  /* Dump block.          */
}


dumpblk(buffer, size, offset)           /* Dump block in hex and ascii.     */
int size;
char buffer[];
int offset;
{
  int thissize;                         /* Define local variables.          */
  int c, i, j, cpos;

  for (i = 0; i < (size  / 16); i++) {     /* Loop for number of lines.     */
    printf("%06x:  ", (i * 16) + offset);  /* Print address.                */
    cpos = 10;

    if (((size/16) - 1) == i) {             /* Check if on last block.     */
      if ((thissize = (size & 15)) == 0)
        thissize = 16;
    } else

      thissize = 16;

    for (j = 0; j < thissize; j++) {  /* Loop for characters in buf.       */
      printf("%02x ", (unsigned char)buffer[i * 16 + j]);
      cpos += 3;

      if ((j == 3) || (j == 7) || (j == 11)) {  /* Do we need a space?    */
        printf(" ");
        cpos++;
      }
    }

    for (j = cpos; j <= 61; j++)        /* Pad line to same length.         */
      printf(" ");

    for (j = 0; j < thissize; j++) {  /* Loop for characters in buf.       */

      if (buffer[i * 16 + j] >= ' ' && buffer[i * 16 + j] <= '~')
        printf("%c", buffer[i * 16 + j]);
      else
        printf(".");

    }
    printf("\n");
  }
}
```

```
int find_in(pathname, dir_req)          /* Find inode of file in directory. */
char *pathname;                         /* If dir_req != 0 then inode must  */
int dir_req;                            /* be a directory.                  */
{
  INODE dirnode;                        /* Define local variables.          */
  char thisname[25], *pos;
  int i, this_in, thislen;

  pos = pathname;                       /* Copy pointer to file pathname.   */

  if (*pos == '/')                      /* If pathname starts with '/'.     */
    this_in = 1;                        /* Read root directory's inode.     */
  else                                  /* Otherwise,                       */
    this_in = curdir;                   /* Read first directory's inode.    */

  read_in(&dirnode, this_in);           /* Read the directory's inode.      */

  while(*pos) {                         /* Loop until end of pathname.      */

    if (*pos == '/')                    /* If we are on a '/' then ....     */
      while (*pos == '/')               /* Skip this and any extra slashes. */
        ++pos;

    if ((thislen = index(pos, "/")) == ERR)  /* Are we at the end?          */
      break;                            /* Yes then we are done.            */

    if (thislen > 24) {                 /* Is length of name too long?      */
      pos[thislen] = '\0';              /* Yes then print error.            */
      fprintf(stderr," CFSU: Name '%s' in '%s' too long.\n", pos, pathname);
      return(0);                        /* Return error.                    */
    }

    for (i = 0; i <= 25; i++)           /* Clear out name array.            */
      thisname[i] = '\0';               /* Also makes sure of null term.    */

    strncpy(thisname, pos, thislen);/* Copy this name from path.            */
    pos += thislen;                     /* Move pointer to new end.         */

    if (strcmp(thisname, "^") == 0) {
      read_in(&dirnode, this_in);       /* Read the directory's inode.      */
      this_in = dirnode.i_parent;       /* Search the parent directory.     */
    }
    else {
      read_in(&dirnode, this_in);     /* Read the directory's inode.        */
      this_in = search_d(&dirnode, thisname);  /* Search current dir.       */
    }

    if (*pos == '\0')
      break;
  }                                     /* End of path and main while loop. */

  if (dir_req != 0) {                   /* Check for directory file.        */
    if ((dirnode.i_stat & is_type) != is_direct)  /* Is it a directory.     */
      return(0);                        /* If not dir then return error.    */
  }

  return(this_in);                      /* Return inode number of match.    */
}


int search_d(dirinp, name)              /* Search directory for file name.  */
INODE *dirinp;
char *name;
{
  int i, b, blocks;                     /* Define local variables.          */
  DIRENTRY dirblock[NUMDIRS];

  blocks = no_blocks(dirinp->i_size);  /* Get the number of blocks.         */

  for (b = 0; b < blocks; ++b) {     /* Loop for number of blocks.          */

    readiblk(dirblock, dirinp, b);  /* Read indirect directory blocks.     */

    for (i = 0; i < NUMDIRS; ++i) { /* Loop for number of dir entrys.      */

      if ((strncmp(dirblock[i].d_name, name, 24) == 0) &&
          ((dirblock[i].d_stat & ds_alloc) != 0)) /* Check for match.      */
        return(dirblock[i].d_inode);  /* Return inode number of match.     */
    }
  }
  return(0);                            /* Else return 0 if not found.      */
}


/*************************************************************************/
/*                                                                       */
/*    READIBLK: Read direct and indirect blocks from inode tables.       */
/*                                                                       */
/*    Function to read the given block from an inode's table,            */
/*    It is (for now) limited to tertiary lookup.                        */
/*                                                                       */
/*    Possible conditions:                                               */
/*                                                                       */
/*       A.  Primary (0 .. 16)                                           */
/*             1.1 -> block.                                             */
/*       B.  Secondary (17 .. 16 + 128)                                  */
/*             2.1 -> 2.2 -> block                                       */
/*       C.  Tertiary (17 + 128 .. ??)                                   */
/*             3.1 -> 3.2 -> 3.3 -> block                                */
/*                                                                       */
/*    All of the range tests for lookup level subtract one from the      */
/*    block number before testing.  This is because the block number     */
/*    is base zero, but the SIZE macros are ordinal (1, 2, 3 ...).       */
/*                                                                       */
/*************************************************************************/

#derine SIZE_1ST 16
#define PTR_2ND 16
#derine SIZE_2ND 128
#define PTR_3RD 17

int readiblk(buffer, inp, blk)          /* Read indirect inode blocks.      */
int blk;
INODE *inp;
int buffer[];
{
  int *indexptr;                        /* Define local variables.          */
  int i, iblk = blk;

  indexptr = inp->i_index;              /* Set pointer to inode's index.    */

  if (iblk > SIZE_1ST + SIZE_2ND - 1) {    /* Tertiary lookup?              */
    readblk(buffer, indexptr[PTR_3RD]);    /* Get table 3.2                 */
    iblk -= SIZE_1ST + SIZE_2ND;        /* Scale down for lookup.           */
    indexptr = buffer;                  /* Our index is in the buffer.      */
    readblk(buffer, indexptr[iblk >> 7]);  /* Get table 3.3                 */
    iblk &= 127;                        /* Strip high bits for lookup.      */
  }
```

```
  else if (iblk > SIZE_1ST - 1) {   /* Secondary lookup?              */
    readblk(buffer, indexptr[PTR_2ND]);    /* Get table 2.2          */
    iblk -= SIZE_1ST;               /* Scale down for lookup.        */
    indexptr = buffer;              /* The index is in the buffer.   */
  }

  if (indexptr[iblk] == 0) {        /* Generate empty block if zero. */
    for (i = 0; i < 512; i++)       /* (For blank compression).      */
      buffer[i] = 0;
    return(512);
  }
  else
    readblk(buffer,indexptr[iblk]); /* Read block from disk.         */
}


int read_in(ibuf, inode_no)         /* Read inode from disk.         */
INODE *ibuf;
int inode_no;
{
  int rrp;                          /* Define local variables.       */
  long pos, rlp;

  --inode_no;                       /* Calculate inode position.     */
  pos = (((fs_istrt + (inode_no >> 2)) * 512) + ((inode_no & 3) << 7));


  if ((rlp = lseek(fd, pos, 0)) == ERR) {  /* Lseek to inode position. */
    fprintf(stderr, " CFSU: Error seeking to inode %d.\n", inode_no);
    return(ERR);                    /* Return error if Lseek trouble. */
  }

  if ((rrp = read(fd, ibuf, 128)) == ERR || rrp < 128)  /* Read inode. */
    fprintf(stderr, "CFSU: Error %d reading inode %d.\n", rrp, ++inode_no);

  return(rrp);
}


int readblk(buffer, block)          /* Read block, return bytes read. */
char *buffer;
int block;
{
  int rp;                           /* Define local variables.       */
  long r;

  if (r = lseek(fd, (long)(block*512), 0) == ERR) {  /* Lseek to block. */
    fprintf(stderr, " CFSU: Error seeking to block %d.\n\n", block);
    return(ERR);                    /* Return error if Lseek trouble. */
  }

  if ((rp = read(fd, buffer, 512)) == ERR || rp < 512) /* Read block.  */
    fprintf(stderr, " CFSU: Error %d reading block %d.\n\n", rp, block);

  return(rp);                       /* Return code from read.        */
}


int no_blocks(fsize)                /* Calculate blocks in file or dir. */
int fsize;
{
  return(fsize == 0) ? 0 : ((fsize - 1) >> 9) + 1;
}


int index(s, t)                     /* Find index to character or the  */
char s[], t[];                      /* end of the string. Return error */
{                                   /* if already at the strings end.  */
  int i = 0;

  if (s[0] == 0)                    /* Check if already at the end.    */
    return(ERR);
  while ((s[i] != t[0]) && (s[i] != '\0')) {
    i++;
  }
  return(i);
}


int strcmp(s, t)                    /* Compare strings.                */
char *s, *t;
{
  for ( ; *s == *t; s++, t++) {
    if (*s == '\0')
      return(0);
  }

  return(*s - *t);
}


strncpy(s, t, l)                    /* Copy fixed length string.       */
char s[], t[];
int l;
{
  int i;

  for (i=0; i < l; i++)
    s[i] = t[i];
}


int strncmp(s, t, l)                /* Compare fixed length string.    */
char *s, *t;
int l;
{
  int i;

  for (i = 0; *s == *t && i < l ; s++, t++, i++) {
    if (*s == '\0' || i == l)
      return(0);
  }
  return(*s-*t);
}


help()                              /* Print help commands.            */
{
  printf("\n These are the available commands for CFSU.\n");
  printf(" All numeric arguements are in decimal.\n\n");
  printf(" Change directory: ........... cd or d <pathname>\n\n");
  printf(" Copy Cromix file to Unix: ... copy or cp <source> [dest]\n\n");
  printf(" Dump disk block: ............ ddump or dd <disk block number>\n\n");
  printf(" List directory: ............. ls or l <pathname>\n\n");
  printf(" Type file to the screen: .... type or ty or cat <pathname>\n\n");
  printf(" Help information: ........... help or h gives this list.\n\n");
}
```



Rick Dhaenens *is currently the Senior Manager for Cromemco, in the Atlanta Regional Office. He has 4 years experience as service department manager in a retail computer store and has been with Cromemco in technical support for 3 years.*

COMMERCIAL MEMBER
Mid United States

SYNERGISTICS INTERNATIONAL LTD.
35 Fountain Square Plaza, Suite 207
Elgin, IL 60120
(312) 695-7775

Full inventory of Cromemco hardware and software. Custom software developed in-house. Vertical market packages available include: Chiropractic Clinics, Architectural Woodwork Job Costing; Social Service Agency Accounting; Auctioneering. Specializing in providing turnkey systems to small and medium sized businesses.

Key Personnel: Jim Knowles, President (Sales)

Major Market Area: Sales: Chicago and suburbs, extending to entire U.S. and the U.K.. Service: Chicago and suburbs.

COMMERCIAL MEMBER
Mid United States

COMPUTER CROSSROADS OF AMERICA, INC.
6 Terrace Shopping Center Richardson, Texas 75081
(214) 231-6108 Twx/Telex 4991118

We are a CROMEMCO MASTER DEALER engaged in DEALER and OEM sales, service and support. We are in the top twenty-five dealers in the U.S. We have a consulting staff comprised of specialists in hardware, software and applications engineering. We are presently engaged in sales from the hardware level (equipment and/or software delivered in an unopened box) through the complete systems level where we take full responsibility for the system hardware configuration, instruction and maintenance of a system. As our name implies THIS IS THE CROSSROADS WHERE IT ALL COMES TOGETHER.

Key Personnel: Ed Fearon, President, Sales & Support
John Rateau, Sales & Support
Danney Jarman, Sales & Support
Joe Essler, Sales & Support
Bill Carnahan, Support

Major Market Area: Sales & Service Worldwide

COMMERCIAL MEMBER
Middle East

MICRO COMPUTER SYSTEMS MARKETING CENTER
P.O Box 1446
Jeddah, Saudi Arabia
(966) (2) 651-7707 or 653-0580
Telex: 928-403068 MICSYS SJ

Authorized Dealer in Jeddah-Saudi Arabia (Western Zone) for Cromemco sales and maintenance of computers, peripherals, software development and design. Strong Arabic Software development.

Key Personnel: Abdul Rahman H. Attar, General Manager
Issam Al Safadi, Administrative Manager
M. Ali Khan, Marketing Executive

SOFT TIPS

SOFT TIPS is a regular column aimed at providing software oriented hints and ideas. Member contributions are encouraged. SOFT TIPS is edited by Norman Vadnais, President of Computer Specialists & Associates, an Orange County Customer Support Specialist. Mr. Vadnais can be reached at (714) 841-3620.

A DIFFERENT ANGLE

At this point, Soft Tips is going to stray from its original direction, to focus on software in general for the new lines of Cromemco Computers. I would appreciate your opinions on both of the points made here and any others you may have. Please forward them to me c/o the I/O News.

CROMIX-PLUS FOR EVERYONE

Boy is it nice when something you have been waiting for for a long time finally arrives. I have had the privilege of playing with version 31.01 of Cromix-Plus for the past two weeks, and I'm in love! Talk about nice. I had heard so many various rumors regarding its availability to WDI users, it was nice to finally get my hands on a copy.

And now for the puzzling points. Rumors have had Cromemco going back and forth recently on their future direction. For a while, Unix was all that mattered. Supposedly the first release of Cromix-Plus was to be the last as far as significant changes go. One line I heard was that Cromemco was now only interested in being the "fastest Unix machines in the marketplace." It was interesting to note the editorial the week after I heard this in one of the major trade publications. It described how Unix vendors were making the same mistakes the original CP/M makers had made, trying to be the biggest and fastest and not worrying about user needs and software applications.

About this same time, Cromemco announced all sorts of applications that were being ported to Unix. Unfortunately, rumor had it none were ever going to make it to Cromix. Time will have to tell on this, since the scuttlebut is that one of the products, Informix-SQL, will soon be ported. What seems interesting, is that so many 'knowledgeable' Cromemco followers have been given so many varying ideas regarding future directions in such a short period of time. I hope this magazine, through an interview or feature article, can clear this up and possibly offer a forum to Cromemco users to voice their desires and have them represented to Mountain View.

PORTING SOFTWARE IS NOT FUN

I have also had the recent pleasure to work with the 3.30 version of Informix. This version of the software offers a lot of new features in the handling of input data, and is really nice to use. But then the problems set in.

I have had many interesting conversations with the technical support department at RDS, the makers of Informix. At first they were very congenial, but then we hit a snag. It seems their standard warranty is only for loading and installing of the software onto your system, and for nothing more! As long as the disks you receive are good and you are able to properly 'brand' their software, the rest is up to you.

If you have a problem running their software, they want you to sign up for their software support package. Good idea, except that after plopping down $1600 for the software they want a few hundred bucks for any support.

Luckily, I was able to convince them the port had legitemate bugs and that they should help me help them to fix it. They are being very helpful, and we should have the port fully working in a very short time. I just wanted to bring up the support policy and see how you all might react to it. Please respond to me if you have an opinion.

MENUS AREN'T FOR EVERYONE

In 1983, MenuWare of Woodside, California, came out with a package to assist in the creation of custom menus for Cromix users. The package was the first of its kind for Cromix users. The idea was a good, if not great, one.

As seems to be standard, the first version of the package that came out was a little rough around the edges. But all in all, the package gave what it was intended to give, a way for a system designer to create simple-to-use menus as a convenient way for anyone to use their computer system. Much feedback was given to MenuWare on possible improvements, and we all heard a new version was coming.

The new version had many new features, but some major problems. When calling MenuWare regarding those problems, little help was forthcoming. The reason: "We are working on our (choose one:) PC-DOS/Unix/other OS version right now. We do not have the time to load up the Cromix version." If I had been nitpicking on some minor problem, then I might understand the brush off. But my problem had major side effects (ALL CURRENT USERS OF MENUWARE PLEASE TAKE NOTE)!

My problem was simple to explain and understand. It occurs if you are using MenuWare's login program in place of Cromemco's. If your user logs in with his name not exactly the way it is in the /etc/passwd file, he is given privileged user status. If this user also has access to screen, ccall, or any other program offering access to the shell, this user can do anything!!! It is a shame such a major flaw in an otherwise useful program causes me to be unable to advise my clients to use it.

BUGS REPORT

I do hope to be receiving more bug sheets from our various members over the next couple of months. We all hate bugs, wish they didn't happen, etc., but we also realize they do happen. Please help to make this column a forum for all bugs, in both new software and old, on C-10's, CDOS systems, Cromix, and Cromix-Plus alike.

A bug forum is important to Cromemco users for two reasons. First, it would be nice to have one place, easily accessible, to verify if your problem is the software's or yours (if you're the first to find it, this won't help you but will greatly help the ones behind you!). Secondly, we should be able, through reader response, to identify which of these little critters is most troublesome to the average user, and can press Cromemco to take care of these problems first. So don't gripe about that bug, WRITE about it instead!!!

VSL International

Continued from front cover

a multi billion dollar giant from Dallas, operating mainly in the oil and gas exploration business. One of the ENSERCH companies, EBASCO, is one of the largest US engineering offices, and is well known for having designed a large number of nuclear power stations.

But lets come back to VSL International. This company is characterized by the following:

— it consists of a large number of companies or branches in different countries, locally operating;

— as a part of marketing it gives the customers extensive support in highly sophisticated analysis and design within the civil and structural engineering field.

I hope that you recognize the discrepancy hidden in these activities. On one side we are not an engineering company, we are not dealing with design, and we cannot have a large engineering staff in every subsidiary because the design work is only an unpaid support to our main business.

On the other side, the requests of the customers are mainly related to very complicated problems, where the solution needs a high level of theoretical sophistication. And as you are probably aware, such engineering is unthinkable without computers. This leads to the following situation:

— we need large, powerful but very specialized computer programs in civil and structural engineering, and because of our narrow specialization. we have to develop and support at least part of these programs on our own. They are not available on the software market.

— due to the characteristics of these calculations (they deal, for example, with non-linear material properties and time dependent phenomena), there is a demand for high performance computing,

— to make the programming support practical, we need compatible computers in all offices,

— to make the cost of hardware bearable also for the smallest of our subsidiaries, we need a low-cost basic unit, which could be extended into a large through-put system.

For a long time there was no solution to this problem, and we used our VSL head office Data General minicomputer or the in-house LOSINGER-owned IBM mainframe. In 1982, when the first professional micros appeared on the market, we started with our CULMANN-Project with the aim of bringing sufficient power to each of our subsidiaries. We made a profound analysis of our demands and thoroughly examined what the market had to offer. Some of our criteria were:

— minimum of 0.2 MIPS computing power

— floating point processor (at least in sight)

— at least 20 bit addressing

— extendability to min. 2 MByte main memory

— recognized (universal) bus

— adaptability of large (100 MByte plus) disk drives

— open to connect third party hardware, mainly peripheral devices like graphic terminals, printers, and plotters

— ½ inch tape unit

— interconnectabilitly of several CPU's

It was not until 1983 that we submitted the results of our investigation: CROMEMCO was the only suitable system.

Today we have installed 12 systems at various places in the world, and I hope that more installations will follow. I must tell you that our subsidiaries decide for themselves if and what system they acquire — we only recommended them to select CROMEMCO. So, for example, our by far largest subsidiary — VSL Corporation in the USA, with 3 main divisions and 9 additional local branch offices, still resists our recommendation. Although the computer is an American one, its selection is made in Europe and this, even inside the same company, is reason enough to be distrusted!

Let me illustrate the range of systems we have installed:

The smallest one was installed a short time ago in South Korea and consists of a 0.5 MByte CS 100-HD 21 with one Modgraph GX 100 graphic terminal and an EPSON FX 80 matrix printer connected both to the computer and to the terminal through a T-switch, to serve as a regular printer, and as a screen hardcopy device too.

The largest configuration is installed in our head office in Berne. It consists of 3 CROMEMCOs interconnected by C-Net. Two of them are CS 200 Systems with XPU, 2 MByte ECC, Maximizer, SMD-controller, 300 MByte Ampex Capricorn Drives. The other is a CS 300 System with 1 MByte ECC, 50 MByte Winchester, 8" and 5¼" floppies and ½" magtape unit. This third CROMEMCO is the utility server for both of the aforementioned number crunchers. It serves a fast Siemens ink jet printer, Graphtec Plotter and some other devices. To keep the systems fast enough we do not use dumb terminals but CORVUS-CONCEPT (also 68000 Motorola-based, ½ MByte) Workstations. In such a way we can off-load all front-end processing to the workstation and use the CROMEMCOs more as high level batch processors.

The interconnection of a large number of CORVUSes (12 installed) with CROMEMCOs caused some headaches because the CORVUS-OMNINET LAN accepts a lot of different systems but not CROMEMCO, and CROMEMCO's C-Net as a truly proprietary LAN accepts only CROMEMCOs. We solved the problem by a third network — English-made CLEARWAY — where the access to the network goes from an RS 232 serial asynchronous port to a network interface unit, which is independent of the connected device. Of course, such a network cannot be used for fast transfer of very big files.

Before concluding this contribution, I would like to vent my imagination:

How will CROMEMCO look in 1986?

— It will be VME-Bus based, with a choice between cheap 68000, faster 68010 and super-fast 68020 with cache memory, where all systems are fully software compatible.

— It will contain an 8086 coprocessor for inside UNIX embedded MSDOS, with PC-slots on this coprocessor board.

— A wide range of preconfigured systems, starting with a small unit priced near a PC, will be offered.

— A smaller, Corvus-like workstation with 14", medium (approx. 600/800) resolution and dual orientation screen will be available.

— A big workstation with 20", high (1000/1200) resolution screen will be available.

— Windowing management software and GKS graphic library equal for both workstations.

— SCSI and extended SMD Interface.

— Support of fast streamers.

— 8 bit parallel, RS 422 and other common interfaces including software drivers.

— Automatic power failure case backup and recovery.

— Truly programmable I/O coprocessors with their own memory.

— Laser disks.

— High resolution document scanner unit with compression software (CCITT-Standard).

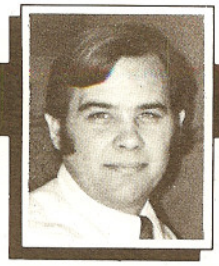
— Full Modula 2 and ADA compilers.

— Extensive self-test routines for all supported hardware will be available, and of course all the units, features and software of today.

In the face of such on open system not one of todays Cromemco system integrators will be inclined to change to Micro-VAX, Convergent or Sun. I hope that at least a part of my imagination will come to pass, and that improved marketing and support will enable Cromemco to grow from the industry's secret brand to a more widely recognized and appreciated one.

About the Author:

Igor Uherkovich is the Junior Vice President of VSL International Ltd., Berne, Switzerland



TEC TIPS

TEC TIPS is a regular column aimed at providing hints for keeping systems up and running. It will not attempt to deal with specific engineering applications or non-standard configurations. TEC TIPS is edited by Richard Quinn, owner of QUINN TEAM, a Southern California computer service firm. Telephone (818) 889-4819

CTD Tape Drive System

For a long time the only way to backup a system was to use floppies. Then the 9 track system — the TDS, came along. What are the relative advantages or disadvantages of each system? Let's investigate.

Using floppies to backup has several big advantages and one big disadvantage. Advantages include portability between systems, low-cost media (the floppy is everywhere and cheap), and low-cost drives. The one big disadvantage is capacity: they don't hold much. To backup a full 20 meg drive takes around 60 5'' diskettes! Because of the time it takes and the hassle of splitting files over disks, people don't backup (if you can believe that!).

The next item to show up was the TDS. It too has several advantages and only one big disadvantage. It is a standard format, using the IBM 9 track tape format, uses readily available 1/2 inch reel-to-reel tapes, and is highly transportable between most 9 track systems, including mainframes. If you are using a large UNIX machine or need to move data or programs from another non-UNIX or Cromemco machine, the 9 track is best. 9 track systems can also hold a lot on one tape and make a great back-up for large hard disk drives, streaming all data rather than a file oriented backup. But if you want, you can easily do a file-by-file backup.

The big disadvantage with 9 track is cost. It is by far the most expensive of all backup systems. It is also big. The Cypher drive used by Cromemco is one of the best but not at all portable. However, service is nation wide and the unit itself is very reliable. The only other problem is that the TDS requires 2 slots on the bus, one for the IOP and another for the CSP. That may be a problem for smaller System 1's or 100's.

The newest member of the backup family is the CTD — which is also currently the most popular. It is fast, can hold a fair amount on one tape, and is low-cost. It also uses the floppy disk controller card as its interface so that no additional cards are needed on the bus.

All installation instructions are for use with the 64FDC card, but in some cases the drive will work fine with 16FDCs. Cromemco does not give any modifications for use with the 16FDC, but anyone who can understand the schematics can figure out what to do. It does take some time and requires study of the card and some trial and error. Your system configuration will dictate the problems you'll have. XPU or DPU system require slight differences. I would love to list those changes but I'm not certain at this writing what they are as Cromemco has changed the CTD software and drivers several times recently and even factory modifications have not always worked in some systems. When everything settles down I'll try to write a complete guide on CTD installations.

The CTD is basically an intelligent tape drive, being microprocessor controlled, that looks like a floppy disk drive to the system. By using a combination of drive select and side select, the tape unit is instructed to read or write the 7 tracks on a tape. The tracks are read or written in a serpentine fashion, going from the beginning to end on the first track, then from end to beginning on the second track, then from beginning to end on the third, and so forth. Hence the tape looks like one long stream to the system, or a disk with lots of sectors.

The CTD suffers from some software/hardware driver problems and is currently undergoing some evolution. We are also learning that the Scotch tapes are not as good as they should be, which has always been my experience with Scotch magnetic media, and I look forward to others beginning to produce this data tape. But in general the tapes stretch causing the tapes to be hard to read. When a tape is inserted in the drive it will automatically be retensioned: run from the beginning to the end without any reading or writing. This lets the tape warm up and gets any bubbles or slack out. I have found more reliable back ups and restores are produced if you repeat this process several times.

Also, some CTDs suffer heat related failures. If you have an early unit without a fan, install one. The drive itself will work better without lock ups. Its easy to install as the opening and grill

are already there. Newer units have fans from the factory.

You can do streams or file oriented backups using any of the above devices. For those who may not understand those terms, a stream is considered a device backup without regard to individual files. If you enter dump /dev/XXX where XXX is the name of a block device like HD0, STD0, STD31, SFDA or so forth, you will see that the system treats the device as if it were one file with a lot of data. Directories, inodes, data, programs, everything comes out in the order it is physically placed on the drive, sector by sector, hence a stream.

A particular file, on the other hand, has nothing but the contents of that file in it. The actual data may be, an probably is, scattered randomly over various sectors on the drive. That means that a file copied will physically be in a different place after it is copied. To backup a single file requires the system to read the directory and gather together the sectors of data from the drive in the proper order to recreate the file, and then write them to the backup device.

All of the above can use streams or file oriented backup. I generally favor streams as they are fast, preserve the directory structure, and require no special knowledge to do. They also will catch all data. (There is nothing worse than restoring a system after a crash only to find that a particular file was never backed up.) On the other hand, file copies will be necessary to transport data between systems or to archive data.

One other point on backup. We are generally talking about hard disk system backup. If you use a floppy-only-system you will generally consider backup to be making duplicates of your floppies from time to time using xfer (CDOS bus systems), copydisk (C-10s), and cptree, rcopy, or copy (Cromix). The TDS 9 track tape will work on any Cromix machine provided you are using the appropriate programs with the version of Cromix you are running. The CTD will work ONLY on Cromix-D and Cromix-Plus and UNIX machines, i.e., only 68000-based hardware.

Floppies can be used on any hard disk system, regardless of operating system or version. However programs like FASTBACK allow you to do streams, prompting for the next floppy with no hassles, and works well for 5 and 10 meg drives. Larger drives are not practical to backup with floppies. Just too many and too long. Cromix/UNIX machines can also use a FASTBACK-like product or tarz80, tar, cptree, copy, or rcopy. Each has advantages or disadvantages depending on what you want to do.

One thing is for certain. If you don't do some type of backup, the question is not if you will lose data at some point but when will you lose data. Figure the best way to backup for your system and do it!!

How Far Can A Cable be Run?

I am frequently asked how far a CRT terminal can be run from the system. I also get the same questions asked regarding printers. Here are some short answers to that question.

Terminals and printers that are serial use what is known as the RS-232 standard. That is simply a definition of voltage levels for a mark and space, or 1 and 0 data bit. Serial information is sent one data bit after the other to reform the 8-bit byte. This type of interface can generally be run much further than a parallel type interface.

In fact I have run serial RS-232 lines over a mile without problems. The factors that determine the maximum distance (which is usually listed as 100's of feet — not miles) is as follows: a) the type of data cable being used, b) the quality and placement of grounds and, c) the baud rate.

For maximum or electronically "noisy" runs, use a shielded cable with the shield grounded at one end ONLY. When I say "grounded" I do not mean to pin 7 of the RS-232 connector. I mean earth ground. The shield MUST be grounded only at one end as grounding at both ends can cause current flow over the shield and failures. Envision the shield as a metal tube to shield against outside electromagnetic noise. Secondly, using two twisted pairs of wires, with one wire from each pair being used as the data ground, will allow the fastest baud rate with the lowest "cross talk" between transmit and receive lines. Such cross talk displays itself in the form of stray characters. If it persists, lower the baud rate and see if the problem clears itself.

The last thing is to be certain that the data ground is not tied to earth ground in the CRT terminal or printer itself. Check the third pole or ground pole of the terminal with pin 7 of the RS-232 connector and be certain they are open or install an isolation transformer on the device without an earth grounded secondary to clear this fault. If you don't, all of the above will be for not.

Generally speaking, parallel data lines are not RS-232 voltage levels and will be subjected to shorter runs because their driver hardware is less powerful. But the techniques described above will still get you much further than the book says you can run.

Card Placement on the Bus

It is often asked if card placement on the bus is critical and generally the answer is no. But I use the following rules of thumb when setting up systems.

First locate I/O cards with cables where they are easiest to connect. That means left in System 2's, to the back in System 3's, 300's and 400's, at the bottom of System 1's and 100's. I put the I/O cards in the order that the priority interrupt cable connects for convenience.

Next, I place the highest power consumers in the middle of the bus as that is where cooling is best and power distribution is best. The DC power comes into the 21 slot mother boards in the middle and distributes left and right. Cards like the Maximizer and memories use more power and deserve to be closest to the source so that slight voltage drops on the mother board will not be a factor in reliable operation.

Thirdly, I place the lowest power consumers to the outside or ends of the card cage. The exception is the CPU card which I always stick in the middle, more out of habit than anything else.

Lastly, I space the cards evenly with open slots between each card where possible. This aids cooling and can make a big difference in older or heavily loaded systems. Generally, System 3A's, 300's, 1's, and 100's will cool well in all configurations if the ambient temperature is in the 65F to 75F degree range. But older System 3's and 2's may have problems.

Most systems could care less where you put anything. I follow the above mentioned guidelines to guard against marginal conditions and just find it easy to follow this pattern. Also, it eliminates a lot of strange cable tangles and makes swapping cards that much easier. Its up to you. CD

INSIDE CROMIX

INSIDE CROMIX is an open forum on both eight-bit and 16-bit versions of Cromix. The subject matter is directed towards helping Cromix users derive more from their systems. Members' contributions are invited. INSIDE CROMIX is edited by Jordan Siedband, who can be reached at 5017 Fairview Lane, Skokie, IL 60077, (312)674-1175.

Publisher's Note:

This issue marks the debut of Jordan Siedband as the editor of this column. Any help, suggestions, complaints, poison pen letters, should be forwarded directly to him: Jordan Siedband, 5017 Fairview Lane, Skokie, IL 60077 (312)674-1175. *If you want your entry returned, send enough postage and a mailing label to yourself before your ideas are ignored or discarded.*

THE "UNIXIFICATION" OF CROMIX

One of the biggest sales points for Cromix was that it would be an honest to goodness workable simulator of the UNIX system. Many of the utilities are directly identifiable with UNIX utilities with different names. Some are exact analogues. In a previous I/O NEWS, (Vol 4, No.4), we submitted a program called delay, which was an almost exact simulator of at in the UNIX system. One of my clients started reading one of the UNIX introductions on the market and exclaimed that he would like the UNIX utility calendar. Brashly, I stated that not only would I do it, but for the heck of it, he could have it free — just to show that I could do anything (well, almost). The following program, written in 68000C will emulate the calendar utility almost exactly. Should you be too lazy to compile it yourself and want a calendar.hlp file for the /usr/help directory as well, send an initialized disk (5'' or 8'') with return postage and a mailing label to the address above, and I will send you copies.

To run the calendar facility, three things are necessary: (1) calendar.bin in the /bin directory; (2) the file calendar in your home directory; (3) the file .startup.cmd in your home directory. The file .startup.cmd should contain at least one line entry, viz., calendar 1

On the first login, the calendar will show. On subsequent logins on the same day the calendar will not be displayed. The calendar may, however, be shown at any time by typing calendar.

The file calendar is a text file and must be totally maintained by the user, as is the UNIX version. Entries consist of a date in the form 12/15/85 and followed by a tab character. If the next line is a continuation, do not enter a date, but do enter a tab as the first character. A typical set of entries is shown below:

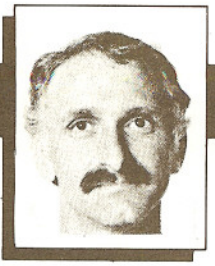
```
 9/15/85  First test date
10/01/85  Second date
          Should print without date
11/10/85  10:30 dentist appointment
11/15/85  good date to show
12/17/85  wait for Robert E. Lee
12/19/85  Test
          Second Line of Test
12/25/85  Happy Birthday
 1/20/86  Noon  Lunch with M. Mouse
 1/28/86  Happy Birthday
 3/18/86  Get new car
```

You must add and remove entries by yourself, as the system will not add or eliminate them for you. The screen editor is an effective way of maintaining that file.

We want more of these utilities from you. Really extensive programs are worth charging money, but short utilities are more fun to share. The more you share these ideas, the better we all become. Good programming and documentation are what we all seek. Let us all share ideas — keep those letters pouring in!!

```
calendar.c  November 9,1985 -  9:50 PM  Page  1

  1           /*      calendar: read calendar file            */
  2           /*      J.Siedband (312)674-1175   10/31/85     */
  3
  4   #include <stdio.h>
  5   #include <modeequ.h>
  6
  7   #define MAXSIZE         256
  8   #define BELL            7       /* ascii bell                   */
  9   #define st_taccess      13      /* time last accessed           */
 10
 11           /* number of days in each month--- to heck with leap year!!  */
 12   int mo[12]={31,28,31,30,31,30,31,31,30,31,30,31};
 13
 14   struct
 15   {                               /* date and time in status      */
 16           char    year;           /* Year number (0 .. 99)        */
 17           char    month;          /* Month number (1 .. 12)       */
 18           char    day;            /* Day number (1 .. 31)         */
 19           char    hour;           /* Hour number (0 .. 23)        */
 20           char    minute;         /* Minute number (0 .. 59)      */
 21           char    second;         /* Second number (0 .. 59)      */
 22   } *st_time;
 23
 24
 25   char    *versn="\375\355\375\355\0\0\0\0\0\0\
 26   calendar: read calendar file     10/31/85\r\n";
 27
 28   char    buff[MAXSIZE];
 29   char    *filename="calendar";
 30   char    *ctl="%s";
 31   char    clrall[6]="\033E";      /* replace with screen clear string */
 32   char    clreop[6]="\033J";      /* replace with clear to eop string */
 33   char    crlead[6]="\033F";      /*   cursor leadin, 1B 45 0 for C5  */
 34   char    offset=' ';             /*  cursor offset, shown for C5     */
 35   long    today,tomorrow;
 36
 37   char    dat[20];
 38   char    *month[]={"Jan.","Feb.","Mar.","Apr.","May ","June","July","Aug.",
 39                     "Sep.","Oct.","Nov.","Dec."};
 40   char    *wkday[]={"Sun","Mon","Tue","Wed","Thu","Fri","Sat"};
 41
 42   main(argc,argv)
 43   int  argc;
 44   char *argv[];
 45   {
 46           int     c,fn,i,j,k,l;
 47           char    date[4],*timstr();
 48           long    tt,dd,rdat();
 49
 50           getdate(date);
 51           i=date[0];      /*  month       */
 52           j=date[1];      /*   day        */
 53           k=date[2];      /*  year        */
 54           l=date[3];      /*  day of week */
 55           today=100*(100*k+i)+j;          /* convert date to 851013       */
 56           sprintf(dat,"%s, %s%3d, %4d",wkday[l-1],month[i-1],j,1900+k);
 57             /*   dat = system date  [18 char]  like "Sun, Oct. 13, 1985"  */
 58
 59                           /*  ======== get tomorrow's date =======  */
 60           if (l>5)        /*    if today is Friday or Saturday      */
 61           {               /* tomorrow must be the following Monday  */
 62                   j += (9-l);
 63           }
 64           else    j += 1; /* otherwise, next day is today +1        */
 65           if (j > mo[i-1]) /* if in next month */
 66           {
 67                   j -= mo[i-1];   /* subtract off # of days last mo */
 68                   if(++i >12)     /* month + 1 & see if past December */
 69                   {  /* if yes  */
 70                           i=1;    /*  Make it January  */
 71                           ++k;    /*     Next Year     */
 72                   }
 73           }
 74           tomorrow=100*(100*k+i)+j;       /* tomorrow like  851014  */
 75
 76           if((fn=open(filename,0))==EOF)          /* read only    */
 77           {
 78                   printf("\nCAN NOT FIND CALENDAR FILE%c\n",BELL);
 79                   exit(1);
 80           }
 81           cstat(fn,st_taccess,st_time);   /* see if already accessed today */
 82           close(fn);
 83           i=st_time->year;
 84           j=st_time->month;
 85           k=st_time->day;
 86
 87                   /*   if already accessed, and any other parameter, get out  */
 88                   /*   This means that if you call calendar 1     and the     */
 89                   /*   file has already been accessed today, exit the prog    */
 90                   /*   If you just type calendar   the program will run since */
 91                   /*   argc = 1 under the last call.                          */
 92           if (argc >1 && 100*(100*i+j)+k == today)        exit(1);
 93
 94
 95           fn=open(filename,0);    /* open the calendar file, read only    */
 96
 97           printf(ctl,clrall);     /*  clear the screen    */
```

Continued on page 33



C-10 ENCOUNTERS

C-10 ENCOUNTERS is a regular column directed to users of Cromemco's personal computer, the C-10. It is edited by Dr. Tom Beer, of Applied Environmetrics, located at 118 Gordon St., Balwyn, Victoria 3103, Australia. Dr. Beer can be reached by phone during business hours at (03) 817-2571. Submit editorial directly to Dr. Beer.

CROS

(The Cromemco Resident Operating System)
or Self-Testing your C-10
and Examining disk details being the confessions of a humbled hacker

Hubris is a word of greek etymology. It describes pride of such overweaning arrogance that it invites retribution from the gods. Well about a year ago now the deity that controls CFD disk drives decided that writing "...my disk drive has now performed flawlessly for a year ..." (I/O News Volume 4 No. 2) was reason enough to zap me. At the same time that those words appeared in print my sole disk drive graunched its last graunch.

At the time, I did not know what was wrong, because the little red light came on, the motor started, graunches did actually emanate from the little plastic box, but then silence—with nothing on the screen. My first thought was that the screen had packed it in but by turning the C-10 off, turning it on and then pressing the ESC key the screen lit up with:

```
Keyboard USA1 Version 07
Cromemco CROS, Version 01.05.
```

So the problem was not the screen. Suspicion immediately devolved upon the disk drive but yet it was working — the lights and noises seemed to prove that. Maybe CDOS.COM or MENU.COM were corrupted on the disk? No. A new disk did not boot any better.

Back to CROS. It has a self test. So I type T. What happens — Memory all OK. Then follow Read/Write tests. Execute the Read test and get the message Data Read OK. Now I know that the screen subsequently blanks after exactly one, and only one, graunch. So I tried to read in the boot track BOOT.SYS by hand. The screen looked like this:

```
;AS                    (Set drive A)
AS;
```

the change of prompt indicated that at least the disk drive was connected. On other occassions I have had disconnected cables on the drive in which case CROS returns with Drive not Connected.

To now read in a disk we need to set the side,track and sector. The default is side 0, and then

```
AS;S 0                 (Set track 0)
AS;RD 100 180 1        (Read into memory location 100)
      0001 0 0002 0    (Two sectors read in)
AS;G 100               (Doesn't do anything)
```

The more observant of you will have been totally unsurprised that nothing happe ed. The boot track is spread over 18 sectors of track 0 with each sector being 128 bytes so that reading only the first sector or two is hardly enough with which to get a running program. What was interesting was that after this little attempt I then tried to boot;

```
;B
```

and the system returned with the message: Disk Controller NOT OK and then stopped. That definitely looked bad. But if I turned the machine on and went straight into CROS and issued the B command to Boot the system then I did not get the error message.

Whilst experimenting I found that the command given in the user manuals ;S 1C 1 (given in Appendix B of the early User manual) or ;S 1C A (given in the C10 Technical manual) do not work. S 1C will work and set track 1C but the next letter or number is supposed to pick the side. Instead one has to use the set side command:

```
;SS 1
```

whereupon the command

```
;S 1C immediately gave ERROR = S 00.
```

Thus side 1 of my disk was not being read.

This was getting interesting. It implied that the CROS read test only bothers to check that side 0 of the disk is properly read — and hence will not pick a problem with reads or writes to side 1. In addition it implied that some of CDOS.COM resided on side 1. Obviously the organization of a CDOS small floppy disk is quite complicated.

The boot track, as mentioned above, is unique. It is set up as a single density track on track 0 of the disk. Track 1 is far more interesting because it is here that you will find the disk directory. To read the directory into memory at memory location 100H use:

```
;S 1
;RD 100 S 2304 1
```

and then use the DM command to examine the structure of the disk directory. The C-10 technical manual explains that the directory consists of 32 byte directory entries. The first directory entry is a disk label with the first byte 81H being used to denote the label. There are 8 bytes for the disk name, 3 bytes to date the disk, then the value 10H to indicate that C10 disks are in 2K clusters. There are then two spare bytes before the number 20H which indicates that C10 disks have a directory composed of 32 (i.e. 20H) 128 byte records. Finally the remaining bits are used to hold the cluster number of the directory and will be 00 and 01 for C10 5.25 inch disks.

The presence of this CDOS label as the first directory entry causes a certain amount of grief. There are many programmers around who are trying to write disk formatter programs so that people can write C10 disks on other personal computers. I have had at least three supposed CDOS double sided double density disks come to grief in my C-10. I only examined one in detail (using CROS) but I noticed that the directory had no label, and the first directory entry was to a program file. This, of course, would confuse CDOS no end.

The rest of the directory consists of file entries. The first byte is used for the file attributes (with E5H denoting a deleted file). The filename and its extension comprise the next eleven bytes with the next byte being the extent number. This indicates the number of directory entries if the file is larger than 16K. Perhaps you have noticed that when you list the directory with the DIR command you may obtain under the list of files the

comment 7 Files, 9 Entries. This indicates that one or two files have multiple directory entries. The last byte on the top row indicates the record count — the number of 128 byte records there are in this entry. Whereas the second row of bytes gives the actual cluster numbers comprising the file. This is basically a list of the chunks of the disk on which the program resides.

With my injured disk drive and CROS I then tried to work out the cluster/side/track/sector map for C-10 disks, remembering that there are 10 sectors of 512 bytes each on a track. (Track 0 is different). Here are my results:

Cluster	Track	Side	Sectors
00	1	0	1,5,9,3
01	1	0	7,2,6,A
02	1	0;1	4,8,1,5
03	1	1	9,3,7,2
04	1	1	6,A,4,8
05	2	0	1,5,9,3

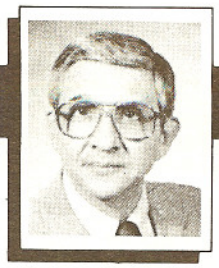
and so on. Obviously the deduced structure for side 1 was guesswork but I checked it out when my drive was finally fixed and the above table seems to be correct. I have tried to check it against the formulae given on page 38 of I/O News Volume 4 number 3 but I confess to total inability to understand the formulae.

Those of you who have followed me this far will now realize why the result of an RD command looks so funny. When CROS executes the RD command it reads in sector 1,2,3,4 etc in that order. But the second 512 byte chunk of the directory is actually in sector 5, whereas sector 4 — being the first bit of cluster 2, is actually the start of the first program file. This crazy way of arranging a floppy disk is known as *sector skew* and you can see that the C-10 has a sector skew of 4. Getting the sector skew wrong is the other reason why disks cannot be read by the C-10 even though the supplier swears that they are Cromemco double sided double density data disks. Each computer manufacturer seems to have his own choice of sector size, sector skew and sectors per track for 5.25 inch disks. Most annoying.

In correspondence with others I have pointed out how one can use CROS to *undelete* an accidentally deleted file. I also pointed out that there are traps and pitfalls. To avoid people falling into such traps or pits I have written a super-duper CDOS undeleter which checks everything about a file and if it is at all possible to properly recover a file it will do so. This program resides on Volume 3 of the Best of Public Domain Software for the C-10 which is available from Applied Environmetrics for US$25, along with other useful utilities such as disk cataloguing programs, file squeezers and encrypters, library creation programs, and more. Included is a program called READBOOT which is an assembly language program to read the boot track off a disk and either write it as the boot track of another disk or store it as a file on your disk to be examined at leisure with ZDUMP, a hexadecimal dump program which is also on the PDS3 utility disk. This little program is very useful if you receive a disk with programs on it that were written on a Cromemco system other than the C-10. If you are running your C-10 under Release 2,3 or 4 CDOS then you cannot boot with such a disk.

Making a disk bootable under CDOS version C2.xx (i.e. Release 2,3, or 4) consists of putting the boot track on track 0 and of ensuring that CDOS.COM is on the disk — because the boot program hunts through the disk for CDOS.COM. It is easy enough to put CDOS.COM on the disk — you copyfile it over from the System disk. But transferring the boot track from the System disk to the new disk requires a program like READBOOT. Alternatively you could do it via CROS by reading in the whole boot track, changing disks, resetting the disk drive head to the beginning of the new disk and using the WD command to write to the new disks.

These are the uses that I have so far found for CROS. Self testing the C-10, transferring system files, undeleting files, and examining strange disks that people send to me. If you have some other interesting or novel use for CROS then I would like to hear about it.



32K CLASSROOM

32K CLASSROOM is a regular column aimed at explaining programming techniques using Cromemco Structured BASIC. It is edited by Bernie Thomas, President of Jakes Manufacturing Corp., P.O. Box 23050, Nashville, TN 37202. Submit any editorial contributions to I/O NEWS in care of the 32K CLASSROOM.

A KSAM MANUAL — PART III

In the first two parts of this manual, we created two KSAM files which we called Names1.kat and Names2.kat. Names1.kat contains names, addresses, etc., with a four-byte numeric KEY. Names2.kat has only one four-byte field in each record which contains the KEY to Names1.kat. Using the following program, we can print a list of names in ascending zip code order.

```
100   Dim Key$(3),Names$(29),Add$(29),City$(19)
105   Dim State$(1),Zip$(9),Phone$(9)
110   Kopen\1\"Names1.kat"
120   Kopen\2\"Names2.kat"
200 *Loop
210   On Error Goto Finish
220   Kgetfwd\2\Key$
225   On Error Stop
230   Kgetkey\1,Key$\Name$,Add$,City$,State$,Zip$
240   @ Name$
250   @ Add$
260   @ City$;", ";State$;" ";Zip$
270   @ : @ : Goto Loop
300 *Finish : On Error Stop : Close
```

We will now discuss **alternate files**. Lets suppose we want to find all of the names in our file who live in California. We could, of course, examine each record to find this information, but there is a much faster and easier way to do this. First, we create an alternate file to our main file. Looking at the structure of our file, we determine that the beginning of the two byte 'state field' is byte number 80. We determine this by adding the number of bytes in each preceeding field. Since Name$ is 30, Add$ is 30, and City$ is 20, State$ begins with the 81st byte; HOWEVER, you must subtract one since Structured BASIC begins its count with zero. The following code will create the alternate file we desire.

```
100   Kopen\1\"Names1.kat"
110   Kaltcreate\1,2,80\"Names11.alt"
120   Kaltopen\2,1\"Names11.alt"
130   On Error Goto Finish
140   Kgetfwd\1\
150   On Error Stop
160   Kaltadd\2\
170   Goto 130
180 *Finish : On Error Stop : Close
```

In order to create an alternate file, the main file must first be opened. In line 110 we have created the alternate, and have given the *channel number* of the main file to which this alternate is associated, '1', the *length* of the alternate key which is '2', and the *beginning byte* of the key which is '80' as we determined above. The rest of the program merely accesses each file and builds our alternate file. Now using the following code, we can find all of the names in any given state.

```
 80   Esc : On Esc Goto Done
 90 *Ask : @ : @ : Input"What State$: ",State$
100   Dim Name$(29)
110   Kopen\1\"Names1.kat"
120   Kaltopen\2,1\"Names11.alt"
130   On Error Goto None
140   Kaltfirst\2,State$\Name$
145   On Error Stop
150   @ Name$
160 *Loop
170   On Error Goto Finish
180   Kaltfwd\2\Name$
190   On Error Stop
200   @ Name$
210   Goto Loop
220 *None : On Error Stop : Close
230   @"THERE ARE NO NAMES LISTED IN ";State$
240   Goto Ask
250 *Finish : On Error Stop
260   @"That's all the names listed in ";State$
270   Goto Ask
280 *Done : Close : Stop
```

You can create many different alternate files to the same main file. The key to the alternate file does not have to be an entire field, and it does not have to include just one field. For example, in line 110 you could say:

```
110   Kaltcreate\1,1,0\"Names12.alt"
```

This would create an alternate file which would key on the first letter of each name.

If you wished to access one particuar record by name, but you did not know the numeric KEY to the record, you would have to examine the records sequentially until you found the name you were seeking. The Kgetapp feature of KSAM is ideal for this type of situation. To demonstrate, lets create another file using Name$ as our KEY and store the four-byte numeric key of the main file in each record. Use the following code to do this:

```
100   Dim Name$(29),Key$(3)
110   Kcreate\4,30\"Names3.kat"
120   Kopen\1\"Names1.kat"
130   Kopen\2\"Names3.kat"
140 *Loop : On Error Goto Finish
150   Kgetfwd\1\Name$
160   Kretrieve\1\Key$
170   On Error Stop
180   On Error Goto Loop : Kadd\2,Name$(-1)\Key$
190   Goto Loop
200 *Finish : On Error Stop : Close
```

Now we can use the following routine to find any particular record quickly and easily by name instead of numeric key.

```
100   Dim Key$(3)
110   Dim Name$(29),Add$(29),City$(19)
115   Dim State$(1),Zip$(9),Phone$(9)
120   Dim Del$(69) : Del$=" "+Del$(-1)
130   Gosub Crt'clear : L=12 : C=5 : Gosub Loc
135   Input"What Name? ) ",Name$
140   If Name$(0,0)="a" And Name$(0,0)<="z" Then  Do
150     Name$(0,0)=Chr$(Asc(Name$(0,0))-32) : Enddo
155   L=12 : C=5 : Gosub Loc : @ Del$
160   Kopen\1\"Names3.kat"
170 *Approx : On Error Stop
180   Kgetapp\1,Name$(-1)\Key$
190   Kretrieve\1\Name$
200   L=12 : C=5 : Gosub Loc : @ Name$
210   L=14 : C=5 : Gosub Loc
220   @ Chr$(7);"IS THIS THE NAME YOU ARE SEEKING? (ans y/n) )";
230   Get\0\I$(0,0)
240   L=14 : C=5 : Gosub Loc : @ Del$
250   If I$(0,0)="y" Or I$(0,0)="Y" Then Goto Found
260   If I$(0,0)#"n" And I$(0,0)#"N" Then Goto 210
270   L=14 : C=5 : Gosub Loc
280   @"ALPHABETICALLY SPEAKING is the name Before or After? ) ";
290   Get\0\I$(0,0)
300   L=14 : C=5 : Gosub Loc : @ Del$;
310   If I$(0,0)="b" Or I$(0,0)="B" Then Goto Back
320   If I$(0,0)="a" Or I$(0,0)="A" Then Goto Forward
330   Goto 270
340 *Back : On Error Goto First : Kgetback\1\Key$ : On Error Stop
350   L=12 : C=5 : Gosub Loc : @ Del$ : Goto 190
360 *Forward : On Error Goto Last : Kgetfwd\1\Key$ : On Error Stop
370   L=12 : C=5 : Gosub Loc : @ Del$ : Goto 190
380 *First : On Error Stop : L=14 : C=5 : Gosub Loc
390   @ Chr$(7);"THIS IS THE FIRST NAME IN THE FILE"
400   Gosub Delay : L=14 : C=5 : Gosub Loc : @ Del$; : Goto 210
410 *Last : On Error Stop : L=14 : C=5 : Gosub Loc
420   @ Chr$(7);"THIS IS THE LAST NAME IN THE FILE"
430   Gosub Delay : L=14 : C=5 : Gosub Loc : @ Del$; : Goto 210
440 *Crt'clear : @ Chr$(27);"+"; : Return
450 *Loc : L=L+31 : C=C+31 : @ Tab(0);"";
455   @ Chr$(27);"="Chr$(L);Chr$(C); : Return
460 *Delay :   For Bell=1 To 5 : @ Chr$(7);
465       For Delay=1 To 500 : Next Delay : Next Bell : Return
470 *Found : Kclose\1\ : Kopen\1\"Names1.kat"
480   Kgetkey\1,Key$\Name$,Add$,City$,State$,Zip$,Phone$
485   Kclose\1\
490   L=12 : C=5+Len(Name$) : Gosub Loc : @"'s KEY is ";Key$
500   @ Tab(4);Add$
510   @ Tab(4);City$;", ";State$;" ";Zip$
520   @ Tab(4);Phone$(0,2);"-";Phone$(3,5);"-";Phone$(6,9)
530   @ : @"PRESS ANY KEY TO CONTINUE ";
540   Get\0\I$(0,0)
550   Run
```

The heart of this program is the Kgetapp command on line 180. The 'app' is short for approximate, and does just what the name implies. It finds the closet match possible, and in most cases, this turns out to be the name you are seeking.

There are two precautions which must be taken in order for the command to work correctly. The manner in which your names are filed must be consistent. In my files, the first letter of each name is upper-case and the following are lower-case. Hence, it is important that the fist byte of the input in line 180 be upper-case. Lines 150 and 155 insure that this is so. Since the length of Name$ will vary, the (-1) must be used following Name$ in line 180.

In line 340, I have used another unusual but useful command: Kgetback. This also does just what the name implies — it moves the record pointer backward to the prior record. It is just the opposite of the Kgetfwd command. A trappable error is generated if you Kgetback beyond the first record, just as the Kgetfwd generates an 'end of file' error.

Please note that the cursor addressing and screen clear escape sequences are for the Model 950 Televideo terminal, and you must, of course, change them to suit you particular terminal.

As always, if you have any questions or suggestions, please contact me.

COMMERCIAL MEMBER
Far East

ASAHI GLASS
Electronics Group
Special Products Marketing Division
1-2 Marunouchi, 2 Chome
Chiyodaku, Tokyo 100
Japan
781-24616 Telex: 24616 ASAGLAS

Complete line of Cromemco hardware and software in inventory. 700 sq. foot training room. Specializing in O.S. modifications. Full service facility, providing technical consulting as well as warranty repair service.

Key Personnel: Shigeo Satoh, General Manager (Systems)
Norimasa Hori, Manager (Sales)
Shinichi Watanabe, Tech/software

Major Market Area: Japan

COMMERCIAL MEMBER
Far East

NCC INTERNATIONAL
Matsunaga Bldg. 1-6-6 Sotokanda Chiyodaku Tokyo 101
Japan
03-(255)7991 / Telex: 781-2523758 KKSHIP J

The oldest microcomputer store of the Byte Inc. Group, offering CROMEMCO to Japan since 1977. This company primarily sells CROMEMCO equipment, and provides high technology and comfortable customer service.

Key Personnel: Kiyotake Ikeda
Ryuichi Kawase

COMMERCIAL MEMBER
Far East

SUPER-NATURE COMPUTER CO., LTD.
29 Lane 300, Jen-ai Road, Sec. 4
Taipei, Taiwan, Republic of China
(02) 705-2442, (02) 700-4858/TELEX: 13937 SNCOMPUT

Our Company primarily sells CROMEMCO computer equipment with high technology and experience, providing installation, warranty repair service and customer education. Specializing in O.S. modifications, software and hardware development for business and industrial applications.

Key Personnel: Miss Su-Chin Kuo, President
Mr. Mark Yeh, Sales manager
Mr. Morgan Chen, Import/Export department
Mr. Ringoi Shiung, Chief of R&D department

COMMERCIAL MEMBER
Australia

MINICOMP
Minicomp Building
104 Mount St.
North Sydney, NSW 2060
Australia
(02) 957-6800 Telex:AA75774 MINICO

Minicomp is a major Australian distributor for Cromemco. Services include installation, integration, software support, professional training and software development. We also offer a wide range of peripherals and software compatible with Cromemco systems. We take great pride in providing fast efficient service and support.

Key Personnel: Mr. Murray Cleworth, Managing Director
Ms. Kim Ballestrin, National Sales Manager
Ms. Lyn Lyons, Software Development

BITS & BYTES

BITS & BYTES is the place to look for the odd bit of information, opinion, programs, profiles and rumors that circulate through The IACU. Our ears are always attuned to any interesting miscellany — if you have something to contribute send it along to I/O NEWS • P.O. Box 17658 • Irvine, CA 92713 • (714) 661-9764

CORRECTIONS & UPDATES

► We experienced a publisher's nightmare last issue when it was discovered (too late) that we had listed an incorrect telephone number in the **Computer Crossroads of America, Inc.** advertisement on page 3. Two consecutive area codes will not work. The correct information follows:

Computer Crossroads of America, Inc.
6 Terrace Shopping Center
Richardson, TX 75081
(214) 231-6108 Telex: 4991118

GPIB INFO

We have received numerous inquiries in regards to utilizing the Cromemco GPIB board. Apparently the accompanying documentation for that board is inadequate. However, combined with the Texas Instrument **TMS9914A General Purpose Interface Bus (GPIB) Contoller Data Manual**, you should have all the detailed information, as well as an example software driver, at hand to satisfy your needs. You can write TI at:

Texas Instruments Incorporated
P.O. Box 809066
Dallas, TX 75380-9990

or call the Division 3 book order number: 1-800-232-3200.

For further information regarding people to contact for help, call or write *I/O NEWS*.

CHICAGO AREA USER GROUP

Jordan Siedband, our new *INSIDE CROMIX* column editor, is interested in forming a local users group for the Chicago area. Anyone interested should contact Jordan at (312) 674-1175.

USER PROFILE — IPS

We are INTERACTIVE PICTURE SYSTEMS, INC., a software design and development team producing personal computer software for the consumer and professional markets. Our strength lies in animation and graphics; our published products include MOVIE MAKER, a real time animation system; OPERATION: FROG, a frog dissection simulation; DANCE, a dance choreography tool; AEROBICS, a home exercise program; and TRAINS, a railroad management game.

So how, you may be wondering, do we use our Cromemcos?

Firstly, our entire office is oriented around our two System 3's and one System 300. All of our source code for all of our programs resides on the Systems. We use a 6502 cross-assembler to produce object code for the appropriate target machine, and then use our own communciations programs to send them directly from the C3's to the target machine.

In addition, even our non-programmers use the Systems. Every desk in both our New York and our Philadelphia offices is equipped with a C-10 terminal connected to one or another of our Systems. They are constantly in use: for word processing, sending and receiving mail, or making entries into our in-house bulletin board where the latest status information for all of our projects is kept.

In order to allow inter-system communications without the expense of connecting to a standard network interface, we have implemented our own program that performs network-like functions, but at a far lower cost (and far lower speed — we run our communications at 9200 baud, rather than the megabaud speeds of a standard network).

These functions include automatic inter-System mail and automatic update of the bulletin boards on all three Systems. To achieve inter-System mail, we had to alter the standard Cromix mail program, but it was well worth the effort. The result is entirely transparent to the user — you simply type "mail [username]", as you would under standard Cromix, and if the user you are specifying is not on your current System, the mail you are sending is forwarded to the appropriate System by our program, and arrives as standard mail there.

We run standard Cromix on all of our Systems. Currently, we are at version 20.63.

We find our Cromemcos to be entirely reliable, and are pleased that we made the decision to centralize our offices with Cromemco hardware.

IPS offers its Cromemco tools for license and sale. For more information, contact:

IPS
42 East 23rd Street, 4th Floor
New York, NY 10010
(212) 475-7053 Telex: 317766.

UNIX/CROMIX WRITEMASTER

It looks like a go on the development of the UNIX/Cromix version of WriteMaster announced in the QUINTEC (now known as QUINN TEAM) advertisement last issue. Richard Quinn said that the response was very positive, and that only a few more pledges are needed. So if you would like to see WriteMaster done right under Cromix or UNIX, give Richard a call at (818) 889-4819.

CROMIX AND UNIX TRAINING

► **MCM Enterprises**, a full service Cromemco Dealership, has completed construction of a training facility in its Palo Alto, CA offices. This new area allows users to take lecture classes or get hands-on training in Cromix and UNIX applications and the operating system characteristics. System Administrator classes are now available. In addition, MCM Enterprises has expanded its trade-in policy to allow owners of older Cromemco systems to upgrade their equipment to newer Cromemco systems with significant savings. If you are interested, call (415) 327-8080 today and ask for Mike Merchant.

► **Cromemco, Inc.** is opening up its training seminars to end-users of their equipment. In the past, these courses were available to Cromemco dealers and service organizations only. For further information contact Cromemco at (415) 964-7400. The calendar for January 1986 appears below.

JANUARY				
MON	TUE	WED	THU	FRI
30 V	31 V	1 H	2	3
6	7	8	9	10
13	14	15	16	17
USE & SERVICE, S-100 SYSTEMS				
20	21	22	23	24
27	28	29	30	31

DPU TO XPU UPGRADE?

Excalibur Computers is interested in learning if there is sufficient demand for a DPU to XPU upgrade, and whether it would be worth $595. If anyone is interested, call Curt at (916) 971-9610.

CONGRATULATIONS HOWARD!

Howard Millman, the Art Director for *I/O NEWS*, and the one most responsible for the magazine's graphic design, received the prestigious Marketing and Merchandising Excellence (MAME) award in a ceremony held this November by the Marketing Council of

Southern California. Millman was honored for having conceptualized and produced the Best Black and White Ad in the "half-page or under" category. His graphic design skills are apparent, and appreciated, in the new look of this issue.

CROMEMCO FIRSTS

All of us who use Cromemco systems have, at one time or another, had to deal with the situation where someone asks you what kind of computer you have, and when you tell them you hear, "CROMEMCO? Never heard of them." Well, next time you hear that you can rattle of a few of the following first-time technical contributions that Cromemco has made to the microcomputer industry:

► Named the S-100 bus, one of the accepted standards for microcomputer bus architecture under the IEEE-696 guideline.

► Developed the first microcomputer system that used the Z-80 microprocessor as the CPU. Subsequently, the Z-80 became one of the most widely used microprocessors for microcomputer systems.

► Developed the first multi-user microcomputer system.

► Developed the first UNIX-like operating system for a microcomputer system (marketed as the Cromix Operating System).

► Developed the first microcomputer system with a Winchester hard disk drive.

► Developed the first complete computer system with 16 megabytes of RAM and 50 megabytes of hard disk space for under $50,000.

► Developed the first microcomputer color graphics system.

► Developed the first microcomputer system with memory addressing beyond 64K bytes.

► Developed the first microcomputer with the capability for IBM Remote Job Entry (RJE) communications, a capability that allowed the microcomputer to talk to a mainframe.

► Developed the first intelligent I/O interfaces by using separate microprocessors on I/O boards.

► First to adapt the mainframe computer concept of I/O Channel processors to microcomputer systems.

► Developed the first microcomputer system that would automatically boot itself from a ROM-resident program, thereby eliminating the complicated, time-consuming operation of front panel indicators and switches.

► Developed the first microcomputer system to include self-programming EPROM capabilities, which allowed the user to add to and/or modify programs by changing the characteristics in the EPROM.

► First to develop a microcomputer system with Error Correcting Memory.

► Developed the first computer graphics system with hardware stenciling capability.

► Developed the first microcomputer graphics program that could synchronize with a television broadcase signal, thereby allowing overlaying of computer signals and television signals.

It was through contributions such as these that Cromemco has pioneered in pushing the mainframe capabilities into the microcomputer area.

LINKER REMEDY

Mr. I.P. Nyffenegger of Geneve, Switzerland writes, "Some years ago I had a program that was linked with an assembled module. The first time I used LINK 3.37 and the second time LINK 3.43. After trying I discovered that the new linker had a bug — it defined two different memory spaces for the same common blocks. Cromemco confirmed the existence of the bug, and until it is fixed, I just use LINK 3.37 with compiled programs. I hope that this solution can help Mr. Photopoulos (INPUT Vol. IV, No. 4)."

Continued ►

Continued

IMI DISK DRIVE REPAIRS

International Memories, Inc. has begun full scale operations as a service and repair center for 5¼ inch and 8 inch Winchester disk drives at its new facilities in Oregon.

The company will emphasize its knowledge and experience based on 5 years in the industry as a manufacturer of high-quality 5¼ inch drives. According to Alan Johnson, Director of Customer Service Repair, the 15,000 sq. ft. facility maintains a full engineering and technical support group, above-industry testing standards and a Class 100 clean room.

In addition, IMI will begin offering service and repair direct to end users of the most leading 10MB drives. It is the first OEM (original equipment manufacturer) to provide this service direct to the public.

The new address for IMI's Customer Service and Repair Department is:

801 C Street
White City, OR 97503
(503) 826-3733 TELEX 284461

FOR SALE

Single-user Cromix Z-80, Z-2H with 3102 terminal & NEC Spinwriter & Wordstar. Call 213-488-9797

For Sale

One FFP..... best offer
call (916) 971-9610

CROMEMCO SYSTEM 3

CS3HD5E, 512K RAM, 1.2 Mb FD, 21Mb HD/STDC, CROMIX 20.63, 7 — C-10 units! 4 5¼" FD drives, dBase II, Calcstar, Datastar, & Wordstar, ... NEC 7700 (3355B) Printer ...
**** $10,425 ****
Call Joe D./Mike T.
Eoff Electric — Portland, Oregon
(503) 222-9411

BACK ISSUES

of I/O News are available

Interested in forming a User Group? Contact I/O NEWS for assistance

COMMERCIAL MEMBER
Australia

INSYSTEMS PTY. LTD.
337 Moray Street
South Melbourne, Victoria 3205
Australia
(03) 690-2899, telex AA30458
84-86 Pacific Highway
St. Leonards, New South Wales 2065
Australia
(02) 439-3788

Australia's largest Cromemco suppliers, with a staff of 18, providing professional services in all areas of computer implementation.

Key Personnel: Dr. Simon Rosenbaum, Managing Director
Norman Rosenbaum, General Manager
Tony Benci, Sales Manager
Ian Holland, Senior Programmer
Sue Stevenson, Sydney Sales

Major Market Area: Australia wide. Dealers in QLD and TAS.

NEW PRODUCTS...

NEW PRODUCTS is a regularly appearing column devoted to announcing and following hardware and software products of interest to Cromemco users. Most information is derived from press releases submitted by vendors. As a result, I/O NEWS cannot be responsible for errors of ommission or any other inaccuracies.

ACUITY SOFTWARE FOR UNIX SYSTEMS

The following summary of the **ACUITY** system was extracted from Issue No. 110 of the Cromemco newsletter, *NEWS for OEM's and ISO's.*

The comprehensive ACUITY system of financial software packages has been ported to Cromemo's UNIX System V computers and is now available to you directly from its developer, **Computer Cognition**.

Computer Cognition, formerly IRISytems, used the MCBA Accounting Software packages as the underlying basis for ACUITY, then reorganized and restructured the code, added numerous enhancements, and transferred the modified packages to the UNIX operating system under a combination of RM Cobol and C.

The ACUITY software includes a comprehensive series of packages that covers a variety of accounting and financial tasks:

Accounts Payable:

An accrual system that allows complete control over disbursements and provides expense analysis reports upon demand.

Accounts Receivable:

A complete package that handles both balance forward and open item customers.

Customer Order Processing:

This System is designed to satisfy the requirements of processing customer orders in wholesale, distribution, or manufacturing environments.

Fixed Assets/Depreciation:

The Fixed Assets system is designed to handle the basic accounting functions for asset acquisition, asset retirement, and the periodic calculation of depreciation of fixed assets for all types of business.

General Ledger:

This module can be interfaced to all other ACUITY systems. A user-defined financial statement report generator is provided to allow the user to specify which accounts and which formats the statement is to include.

Inventory Management:

The Inventory Management package is multiple company/multiple warehouse tool for manufacturing and/or distribution.

Inventory Management/Bill of Materials Processor:

The Bill of Materials Processor keeps track of all the raw materials, parts and subassemblies that are used to create a finished product.

Labor/Other Direct Cost Projections:

ACUITY's Labor and ODC projections systm is a forecasting system to enable the Project Managerto accurately estimate project labor and non-labor costs and revenues, using a spreadsheet interactive format.

Master Scheduling:

This package is designed to provide the necessary tools to plan production schedules.

Material Requirements Planning:

Material Requirements Planning closes the loop on the material planning cycle by utilizing data generated by the Master Schedule, Customer Order Processing, and Purchase Order and Receiving packages.

Project Management (Job Costing):

This program may be used for standalone cost tracking or. when used with the payroll, accounts payable, accounts receivable, and general ledger, as a full job cost system. It is tailor-made for engineering firms and government contractors.

Project Scheduling:

The Project Scheduling (VUE) system is an easy to use online system that combines charting capabilities using critical path methods and project planning.

Purchase Order and Receiving:

This tool provides for the maintenance of purchasing records and can predict cash requirements, monitor vendor performance, and help prevent materials shortages.

Work Breakdown Structure:

Work Breakdown Structure Reporting is used in conjunction with the ACUITY Project Management System and is well suited for government contracting specifications.

Computer Cognition has also announced that three new ACUITY products will be available in the fourth quarter of 1985: Acuity Reportwriter; Production Control; and Acuity Commitments (an accounting package).

The ACUITY series offers many outstanding features, including compatibility, audit trails, release tracking, standards, and a wide scope of accounting system modules.

The ACUITY line is available directly from the developer. For more information or to order, contact:

Sherril Harper
Computer Cognition
225 West 30th Street
National City, CA 92050
Telephone: (619) 474-6745

INFORMIX-SQL ON UNIX & CROMIX

In line with its efforts to increase the base of third-party software available for its UNIX and Cromix-Plus lines of microcomputers, Cromemco, Inc. of Mountain View, CA has announced the availability of the new **INFORMIX-SQL** RDBMS developed by **Relational Database Systems, Inc.** of Palo Alto, CA. The following product description is derived from the RDS product fact sheet.

Informix-SQL is a fully relational database management system for microcomputers. It is based on SQL, the Structured Query Language developed by IBM, which is something of an industry standard in the way of a syntax for accessing and managing data.

Included with the package are a set of tools specifically designed for application building. Incorporated within Informix-SQL are proven and full-featured screen generation utilities which allow for query-by-forms, calculations, and extensive data validation. A complete report writing capability with commands for sorting, grouping, and formatting data as well as performing calculations and aggregate operations is provided by the ACE report generator.

The design and implementation of custom database applications is facilitated by easy-to-use menu creation facilities. Performance is enhanced through the utilization of advanced retrieval techniques. And because it was designed around the Motorola 68000 microprocessor, Informix-SQL can allow for an unlimited number of columns, tables and indexes within a single database.

Embedded SQL and Tools for C

Embedded SQL provides SQL query statements to access databases from C programs. It has the ability to accept dynamic queries. The Tools for C give the ability to access C functions from Perform and Ace, the screen builder and report writer of Informix-SQL, and includes subroutines to perform data type conversion and to manipulate strings.

RDS products are compatible with a growing number of systems including Cromemco, Altos, AT&T, Convergent Technologies, HP, IBM, Plexus, and Sun Microsystems.

For ordering information contact Cromemco, Inc. For further

information regarding the product contact:

Relational Database Systems, Inc.
2471 E. Bayshore Road, Suite 600
Palo Alto, CA 94303
(415) 424-1300 Telex: 467687

NEW PROM PROGRAMMING SOFTWARE

JG ENGINEERING CO., Santa Ana, California has released a new software product designed to facilitate transfer of PROM object code from Cromemco computer systems to DATA I/O System 19 Prom Programmers. The program will also support any of the new DATA I/O systems with compatible serial interface characteristics.

The package creates a 64K virtual memory disk file into which one or more object files are loaded. Under program control PROM blocks of variable size (.5K X 4, 1.0K X 8, 2.0K X 8, 4.0K X 8 or 8.0K X 8) are transmitted to the DATA I/O for PROM programming.

Utility features include: creation and view of 64K or 256 byte memory access map; block-byte dump and/or modify; creation of checksum over specified block of data; and automatic block control for downloading.

The program is supplied on 5 ¼" CDOS diskette with source for drivers included. Extensive operation and installation manual included. Single machine license. Retail price $300.00. Delivery from stock. Dealer discounts available. Contact:

JG ENGINEERING CO.
2617 South Griset Place
Santa Ana, CA 92704
(714) 241-1936 Telex: 5101003041 "JG ENGINE"
EASYLINK MAILBOX #62877291

PSD FILLS THE IBM PC/CROMEMCO GAP

SUNNYVALE, CA — **ProtoMatrix Software Development** (PSD) has just released two products they feel will give them an even bigger edge over the competition in the Cromemco arena. The new products, dubbed **ProCall/PC** and **ProCall/PC plus** are Cromemco compatible communications products with a difference — they run on IBM PC and compatible computers.

PSD started out 4 years ago with a terminal communications product called ProCall for the Cromemco S-100 based computers. The product, at that time, was the only full-featured communications software that offered the popular XMODEM file transfer protocol while taking advantage of the power of Cromemco's Cromix operating system. The product was well received and soon became very popular with users needing to send and receive disk files with other computers. PSD has since released a version of ProCall for the Cromemco C-10 personal computer, and for Cromix-Plus. Now, after almost two years of work, they have released a new communications product called ProCall/PC.

"Now we cover all the bases," says ex-Cromemco engineer Jeff McNaught, who is now at PSD. "We've found that many Cromemco users also have an IBM or compatible hanging around that they use for other tasks, but until now, have not been able to connect the two computers together in any meaningful way. With ProCall/PC, they can connect the IBM to the Cromemco as a 3102 terminal and use all of the Cromemco software they couldn't run before."

ProCall/PC, while being command compatible with its predecessor ProCall, is touted as being vastly enhanced and more user friendly. For instance, instead of mixing program prompts and communications data on the same screen, ProCall/PC uses a status line, windows, and dialog boxes (a la Macintosh) to communicate with the user.

"Being able to sit in front of an IBM and shoot files back and forth between the PC, Cromix, S-100 CDOS, and even C-10 systems easily gives the user an incredible amount of flexibility — not to mention the terminal emulation capability in ProCall/PC plus," says McNaught. ProCall/PC plus includes DEC VT 52, VT-100, and Cromemco 3102 terminal emulations in addition to the features found in ProCall/PC.

ProCall/PC can transfer files with Cromemco systems, and other ProCall users with the PSD PCXFER protocol, and with BBS systems and other computers with the XMODEM protocol.

PSD notes that the features that ProCall users most wanted added have been put into ProCall/PC.

ProCall/PC began shipping October 1st and PSD reports sales have been excellent since recovering from an order backlog caused by a typesetting error (the 100+ page user manual is laser typeset) that delayed shipments.

ProCall/PC is available for 8086, 8088, and 80286 based computers running PC-DOS or MS-DOS version 2.00 and above. ProCall/PC, with TTY terminal capability, file transfers with XMODEM and PCXFER protocols, macros, auto log-on and log-off, modem auto- dialing, and other features is priced at $149. ProCall/PC plus adds several terminal emulations and is priced at $249. PSD supports their products with telephone technical support and a software update service. PSD distributes its products through Cromemco dealers or direct to qualified institutions.

MORE ON PSD'S ProLink

Product Objective

ProLink is a "remote communications facility." It enables the Cromemco C-10 Personal Computer to be operated remotely — over the phone, or directly wired — either by a terminal, or another computer (with appropriate software) from miles away. This function provides access to the computer for executives on the road with a portable terminal, field sales people reporting leads, remote diagnosis of software or hardware problems, remote demonstration of software, or any of hundreds of other applications.

ProLink accomplishes connection to the remote by means of the built-in serial port on the C-10. Either a modem or a terminal may be connected here to support the remote user(s).

With ProLink installed, the local user still has access to all functions available previously.

Speeds Supported

ProLink supports the full range of serial speeds for remote terminals and modems. Speeds of 110, 300, 1200, 2400, 4800 and 9600 bits per second (BPS) are supported. ProLink also supports automatic 300/1200 BPS speed detection when connected to a modem.

Features

When you call your C-10 running ProLink, you get access to the whole computer. You can display directories and run utilities and programs as you need them. You need not be limited to a single program with limited capabilities (Although, if you desire, you can set ProLink to limit remote users to a single program or task — and automatically log them off and hang-up when they finish).

When not being accessed remotely, the C-10's video display functions normally. When a remote user connects, ProLink automatically adjusts the display speed to that set for the remote. When being operated remotely, only the video display speed is affected, and the computer's microprocessor continues operating at full speed.

When a remote user is connected, both the local and the remote screens identically display the computers' activity. Likewise, both local and remote keyboards are active for input. Remote users with compatible terminals or with ProCall/PC (running on an IBM PC or compatible) may also run graphic programs such as Hdtest, SpellMaster, and even Chess.

The ProLink code is super-compact and uses only 2K of memory when loaded. ProLink then remains resident and appears only when needed. You might not even know ProLink was there — except for the new cursor, which lets you know what mode ProLink is in.

ProLink is simple to use. No commands need be entered when loading the program.

File Transfers

If you use our popular ProCall (for Cromemco computers) or ProCall/PC (for IBM PC and compatibles) programs to connect to your C-10, you can enjoy the luxury and peace of mind of error-free binary or ASCII file transfers in both directions.

Security

Prolink provides optional password protection to provide a security gate for remote users wishing to access the system. The local user is unaffected.

You may set ProLink to execute a particular program when a remote user enters the system, and eliminate their access to CDOS.

As a Bulletin Board System

ProLink has a 'BBS mode' which is compatible with BASIC software written for use under 'BYE' — and provides many additional features.

Other Applications

Companies can remotely update and query sales offices progress or needs — even late at night, when no one is in the office.

Software companies may provide automatic updating of users' software. Traveling salesmen can make sales calls with a portable terminal and demo software on-site or provide accurate estimates for jobs.

An office with several executives on the road can establish a low cost remote mail system and keep in touch when it is convenient for each individual. Memos can be downloaded by the field personnel and status reports uploaded to the office "message center" computer.

You can finish up some office work from the comfort of your easy chair using an inexpensive terminal at home. You don't have to drag your computer home with you. The status of long jobs can be monitored remotely without disturbing the job itself.

Vertical software companies can do remote software maintenance or troubleshooting over the phone. The customer support engineer can actually see what is happening on the customer's computer and download and run diagnostics software.

Enterprising dealers can set up a "free demo" system for prospective customers. Customers may even be allowed to download a demo version of the software for even more exposure.

Specifications

Operating Speeds: 110, 300, 1200, 2400, 4800, and 9600 BPS. Auto speed detect for 300/1200 baud.

Connect/Disconnect Interface: Interrupt driven, with visual indication of REMOTE/LOCAL mode.

CRT Interface: Software connected directly to CDOS driver routines. Transmits all characters and legal video functions to display of remote system and local system simultaneously.

Keyboard Interface: Interrupt driven, direct entry to CDOS ROM to allow program interruption and type-ahead functions.

Program Options: Full remote access, Single program only, automatic startup message/commands, Control-C masking, password protection.

Included Utilities: PlMod — sets ProLink operation modes. Time — sets and displays the real time clock. Boot — re-boots the C-10 from the keyboard. C-10 PcXfer — supports error-free binary or ASCII file transfers with remote callers using ProCall or ProCall/PC.

Availability: ProLink is available for Cromemco C-10 Personal Computers with CDOS in ROM (version 3.07). See your dealer for details.

Price: 195.00 U.S. Dealer and Quantity discounts are available. Site Licenses are also available. PSD Part number PLC5.

ProtoMatrix Software Development
12564 Connemara Way Sunnyvale, CA 94087
(408) 749-1292

MEDICAL PRACTICE MANAGEMENT SYSTEM

T.L.C. Research Associates has developed **MPMS**, a comprehensive Medical Practice Management System. The design of MPMS has been guided by the premise that each medical practice is unique with respect to its physical structure, allocation of personnel, accounting practices, record management procedures and a host of other factors. There are, of course, a number of common denominators and global specifications which apply to all practices. In addition to meeting these global requirements MPMS is designed to be readily adaptable to your specifications. Because the software of MPMS has been designed for adaptation to your specific needs you derive the benefits of a custom-developed system at a cost which is competitive with so-called "off the shelf" systems. In contrast to an "off the shelf", system MPMS adapts to your practice, you do not adapt your practice to MPMS.

MPMS is designed for a multi-user, multi-tasking environment in which the various members of the office or clinic staff simultaneously perform specific tasks.

MPMS provides a reliable, cost-effective system organization for the full range of medical office activities including:

- patient demographic management
- subscriber and billing database management
- patient medical database management
- appointment and "drop in" scheduling
- daybook maintenance
- creation and maintenance of practice-specific diagnostic and assessment menus
- automatic updating of patient medical records
- automatic log in of last general assessment, last annual health examination and all visit dates
- billing generation
- account reconciliation
- information retrieval on all demographic and medical database information
- report writing and word processing
- statistical analyses of diagnostic and assessment data
- automated formatting of letters and case summaries.

For further information on MPMS contact:

T.L.C. Research Associates Ltd.
39 Ravenglass Crescent
London, Ontario
Canada N6G 3X7
(519) 679-1481

COMMERCIAL MEMBER
Eastern United States

SYSTEMS ATLANTA, INC.
P.O. Box 99
Highway 5, Toonigh Road
Lebanon, Georgia 30146
(404) 928-0240

As one of Cromemco's oldest dealers, Systems Atlanta is well experienced in hardware and software implementation. With over 1000 systems installed and a full staff of highly seasoned employees, Systems Atlanta offers technical support for operating systems, application software and hardware design. Specific configurations include telecommunications, graphics, data base management as well as fully integrated accounting systems. Systems Atlanta has authored several specific applications packages such as Manufacturing and Inventory Control, Church Management, Job Costing and Unix based programs.

Key Personnel: Charley Dobson, President & G.M.
Betty Dobson, Dir. of Finance & Admin.
Gary Kendrick, Dir. of Marketing
Steve Garrison, Operations Manager

Major Market Area: Worldwide, with exports to South America, Europe, the Middle East and Canada.

COMMERCIAL MEMBER
Canada

D.E. SYSTEMS LTD.
1284 Wellington St.
Ottawa, Ontario
Canada K1Y 3A9
(613) 729-5164

D.E. Systems Ltd. is a full service company offering Cromemco Hardware, Software Development, Education and Application Programs. We have developed integrated Inventory, Point-of-Sale, Invoicing, Accounting and Sales Analysis programs as well as a Courier Package. We specialize in Cromemco Computers for government and small businesses. We have most Cromemco products in stock and offer technical support on the hardware and software. We offer maintenance of all Cromemco equipment and related peripherals.

Key Personnel: Bruno Dugas, President
Keith Corkum, Director (Systems Development)
Dwight Presley, Senior Analyst

Major Market Area: Eastern Canada

CE: New Text Editor

Continued from front cover

proficient in its use. Despite the friendliness, there were some annoyances such as a marked lethargy when editing large files, a lot of screen repaints, a lean command set, and an incompatibility with non-Cromemco terminals (which clever programmers were quick to change).

The Best of Both Worlds

CE combines the more advanced capabilities of vi within the easy-to-use, get-what-you-see framework of Screen. Although CE can be thought of as an enhanced version of Screen, because of the compatibility between screen file formats and CE file formats, it is probably more apt to recognize it as the new child it is — born of Screen and vi. And that puts Cromemco in the envious and commercially advantageous position of having something that other vendors of UNIX machines can't offer. It's one more reason to put Cromemco ahead of the rest.

Dazzling Speed

The people at Cromemco who developed CE wanted it to be fast. That's what they got. The primary reason for this achievement is that it is written in 68000 C code. Other techniques were also employed. For example, CE takes full advantage of the RAM memory resources available in the 68000 environment. The entire file being edited is loaded into RAM memory: the performance killing routine of swapping between RAM and the disk when editing large files is no longer necessary. With CE, when you want to jump around within your 400K file, you give the command to go and, before you can blink, you're there!

By utilizing the intelligent capabilities of the terminal, e.g. line insert, CE minimizes the number of screen repaints. This translates to a more smoothly flowing editing session, and fewer annoying delays.

Efficient Editing Operations

The impressive performance of CE arises from more than mere hardware. Of much greater importance is the set of text manipulating operations you have at your disposal, and the ease and efficiency with which you can utilize them.

Much of the work done in a typical editing session will involve series of repetitive operations: to move about through your text you must repeatedly use the arrow keys; a word can be deleted by repeatedly deleting letters; all occurences of one word can be changed to another by repeatedly searching and replacing. All of this repetion — this pressing of keys — takes time. Therefore, anything that will diminish the amount of work involved in repetitive operations will lead to greater efficiency. With CE, this attention to efficiency is apparent at several operational levels.

For one, consider what's involved just in getting the cursor positioned where you want it on the screen — to insert, delete, or perform some other manipulation. You could spend a lot of time motoring around with the arrow keys, moving character by character. CE enables you to move forward and backward word by word, or to jump directly to the beginning or end of a line, or to preset markers. InDelete mode, you can delete character by character, or from the cursor to the end of the word. This character/word option is also part of the K(ase) command for toggling between upper and lower case.

At a higher level, the need to repeat a command frequently arises, and not having to retype it would be a great convenience. This capability is provided by CE's a(gain) command.

Flexible Operation

Unlike Screen, there are a number of jointly selectable options available for use during an editing session under CE. These can be selected when CE is invoked from the command line as dash options or during the editing session.

The -r option (read-only) does not accept commands to change the contents of the file; without this option, all CE commands are accepted (the -u option, which is assumed).

The -c option terminates all lines with a CR,LF sequence; without this option, lines are terminated with LF only (the -n option, which is assumed).

The -e option does not replace blanks in the output with TABs; without this option, leading blanks are compressed with the appropriate number of TABs (the -t option, which is assumed).

The -v option disables the auto-indentation feature of CE (the -i option is assumed, causing CE to start a new line at the same column position as the previously entered line — a very handy feature for indenting program code or writing outlines).

The -f option displays the name of the file being edited; without this option, the row and column address of the cursor is displayed instead (the -a option, which is assumed).

Sequential editing of files can be accomplished via the -m option and the list of files to edit. Without this option, CE takes the first filename as the input file and the second filename (if any) as the output file (the -s option, which is assumed).

The -m option has a filename as an argument. That file should contain some alternate termcaps data-base, to be used instead of the /etc/termcaps.

Richer Command Set

The foundation of the CE command set is derived from Screen. In fact, most all of the screen commands are present in the same form within CE. However, in most cases, the basic commands have been expanded upon.

CE differentiates between lower case and upper case commands. Most of the new CE commands are selected by the upper case letter of the analogous Screen command. For example, a lower case "i" works the same as in Screen, but an upper case "I" inserts one blank line without pushing the rest of the line after the insertion point.

Other convenience features have been added. For example, the f(ind) command can be made to be case sensitive, as can the s(ubstitute) command. With the w(rite) command, pressing "a" before giving the marker numbers appends the specified text to the end of the named file.

A number of CE commands have no counterpart in screen. With CE, characters typed after the 80th column are not wrapped to the next line, but are "hidden" outside the edge of screen. A horizontal scroll of the editing window is available through the Y(ank) command. This horizontal scrolling feature greatly simplifies the task of putting together documents wider than the standard 80 columns. The lower case variant of the command (yank) allows the user to change the indentation of a portion of the file that is defined by markers. This is especially useful in files with columns, where the indentation of blocks of columns could be changed at one time, rather than with tedious use of the delete function.

C programmers will be particularly happy with the b(racket) command, which jumps to the next matching bracket if the cursor is positioned on a "(", "[", or "{". If there is no matching bracket, the editor beeps and the cursor does not move. This little feature alone can save untold time and effort in program debugging.

Terminal Compatibility

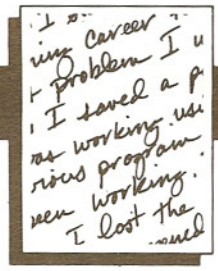
CE makes use of the termcaps capability of UNIX and Cromix-Plus. This feature enables the user to optimize the editor for whatever terminal is in use. A wide selection of terminals are predefined in the editing program; this, combined with the emulation modes, covers the major terminals that are likely to be used.

Migration Between UNIX and Cromix

CE helps bridge the gap between UNIX and Cromix partitions by providing a common means for editing files. Text files created with CE under Cromix can be directly transferred to and operated on by CE running in a UNIX partition. And by virtue of the compatibility between CE and Screen files, files created under Screen can be conveniently transported to a UNIX environment. Operationally, the common ground provided by CE makes it that much easier to switch between working under Cromix and working under UNIX.

SUDS Updates

Speed and efficiency of operation, ease of use, flexibility and powerful capabilities are all brought together in the new CE editor. CE will be sent as a SUDS note to all subscribers to the Cromix-Plus and UNIX System V Software Update Service, and will be included in all shipments of Cromix-Plus and UNIX beginning immediately. The version number that includes the CE editor is Release 3 for UNIX System V and release 5 (version 31.04) of Cromix-Plus.



USER NOTES

USER NOTES are useful techniques, tips, ideas and other helpful information gleaned from our member's experiences with their Cromemco systems. If you have something along these lines that you want to share, write it up and send it to I/O NEWS, c/o USER NOTES, PO BOX 17658, IRVINE, CA 92713.

Dear Editor:

I have enclosed a copy of a program listing for using a non-Cromemco printer, such as C.ITO's dot matrix printer — the CI3500, with Cromemco's WriteMaster program. The short program is written in 'C', and is designed to run on the Cromix operating system. The program is easy to use: after creating a print file such as xxxx.prt with the WriteMaster print command you would simply type the following.

```
% conv < xxxx.prt > xxxx.cit
% spool xxxx.cit
```

I hope that some of your readers will find it useful in gaining full advantage of the WriteMaster program — without having to resort to the expensive 3355A/B daisy wheel printer. However, please note that this conversion program does not include proportional spacing capability, which is one of the significant advantages of the 3355A/B.

Additionally, the program is easy to modify from the original to one that suits any other printer.

I referenced the Cromix Operating System Manual and the 3355A/B driver manual in developing the program.

Sincerely yours,

Tetsujiro Yasushi
3-20-702, Hinode-cho,
Yokosuka-shi,
Kanagawa-ken
238 JAPAN

```
/*
        This program converts writemaster print file to work correctly
        with other manufacturer's printer. (file name "conv.c")

        written by T. Yasushi, 28/10/1985

        USAGE:
                % conv < wrmrfile.prt > wrmrfile.cit
                % spool wrmrfile.cit
           or
                % conv < wrmrfile.prt | spool

        Conversion codes are as follows;

        boldface on : 06h > BOLDON
           "     off: 06h > BOLDOFF

        underline on : 01h > UNDERON
           "       off: 01h > UNDEROFF

        superscript on : 03h > SUPON
        superscript off: 02h > SUPOFF

        subscript on : 02h > SUBON
           "       off: 03h > SUBOFF

        init printer : 7fh > INIT

        set 10ch/in  : 8ch > PICA
            12       : 8ah > ELITE

        set single line space : 19h 88h > LS10
            one and a half    : 19h 8ch > LS15

*/
#include <stdio.h>

#define ON   0
#define OFF -1

/*
        Control codes definitions for CI3500, C.ITO's Dot matrix printer
        CI3500 seems not to support proportional spacing.
*/

#define PICA      "\033[1w" /* set horizontal pitch to 10 CPI */
#define ELITE     "\033[2w" /*                         12 CPI */
#define BOLDON    "\033[93h" /* boldface on */
#define BOLDOFF   "\033[931" /* boldface off */
#define UNDERON   "\033[92h" /* underline on */
#define UNDEROFF "\033[921" /* underline off */
#define LS10      "\033[1z" /* set vertical pitch to 6 LPI (single) */
#define LS15      "\033[6z" /*                       4 LPI (one and a half)*/
#define SUPON     "\033[94!h" /* superscript on */
#define SUPOFF    "\033[941"  /* superscript off */
#define SUBON     "\033[94\"h" /* subscript on */
#define SUBOFF    "\033[941"   /* subscript off */
#define INIT      "\033[2\"z"  /* set letter quality PICA mode */

main()
{
int c,bold,under,supst,subst,intro;

bold = OFF;
under = OFF;
intro = OFF;
supst = OFF;
subst = OFF;

while((c = getchar()) != EOF) {
        if (intro == ON) {
        switch(c) {
        case  0x88  : printf(LS10);
                      intro = OFF;
                      break;
        case  0x8c  : printf(LS15);
                      intro = OFF;
                      break;
        default : ;
        }
        }
        else {
        switch(c) {
        case  0x06  :
              if(bold == ON) { printf(BOLDOFF); bold = OFF;}
              else { printf(BOLDON); bold = ON;}
        break;
        case  0x01  :
              if(under == ON) { printf(UNDEROFF); under = OFF;}
              else { printf(UNDERON); under = ON; }
        break;
        case  0x02  :
              if(subst == ON) { printf(SUBOFF); subst = OFF;}
              else { printf(SUPON); supst = ON; }
        break;
        case  0x03  :
              if(supst == ON) { printf(SUPOFF); supst = OFF;}
              else { printf(SUBON); subst = ON; }
        break;
        case  0x8c  : printf(PICA);
        break;
        case  0x8a  : printf(ELITE);
        break;

        case  0x19  : intro = ON ;
        break;

        case  0x7f  : printf(INIT);
        break;

        default : putchar(c);
        }
        }
```

LOCAL USER GROUPS

Arizona Association of Cromemco Users
Contact: Jo Ann Drake, President
2207 West Eugie Avenue
Phoenix, AZ 85029
(602) 993-9589

Australia User's Group*
Contact: Minicomp
Minicomp Building
104 Mount Street
North Sydney, NSW 2060
Australia
(02) 957-6800
Meets Monthly
*Publishes "Minicomp/Cromemco" a monthly newsletter

Bay Area Cromemco Users & Programmers (BACUP)
Contact: Raymond Barglow or Alan Walworth
United Word & Data Processing
2345 Fulton Street
Berkeley, CA 94704
(415) 841-0708 or (415) 548-2692

Cromemcohorts
Contact: Dr. Brent Lowensohn
4747 Sunset Blvd.
Los Angeles, CA 90027
(213) 667-8972

Cromemco Users' Group of Australia*
Contact: Tony Stringer
52 Beechwood Avenue
Greystanes, 2145
*Publishes a magazine "CROME-SOMA"

Cromemco Users' Group Holland (CUGH)
Contact: Joop Kohler, Secretary
P.O. Box 120
2910 AC Kieuverkerk a/d IJssel
The Netherlands 01803-13300

Cromemco Users' Group*
Contact: Peter Norman
The University of Newcastle Upon Tyne
Department of Chemical Engineering
Merz Court, Claremont Road
Newcastle Upon Tyne NE1 7RU
England
Newcastle 28511, Ext. 3278
*Publishes Cromemco Users' Newsletter (CUG)

Cromemco Users' Group Ontario, Canada
Contact: Lloyd Parker
Hiram Walker Resources, Ltd.
Suite 600
1 First Cnandian Place
Toronto, Ontario
Canada M5X 1A9
(416) 864-3349

Cromemco Users of Orange County, California
Contact: Michael Peterson
Accountability Systems
700 South Tustin Avenue
Suite B
Orange, CA 92667
(714) 639-4570
Meets third Tuesday monthly

Insystems Pty. Ltd.*
Contact: Norman Rosenbaum
337 Moray Street
South Melbourne, Victoria
3205 Australia
(03) 690-2899, Telex: AA30458
*Publishes "Cromemco UPDATE" a bi-monthly newsletter

Illinois Users' Group
Contact: Jim Knowles
P.O. Box 631
Elgin, IL 60120
(312) 695-7775

Indonesian Cromemco Users' Group (ICUG)*
Contact: Zafir M.A. Pontoh
Computation Lab
Department of Regional & City Planning
Bandung Institute of Technology
10 Ganesha
Bandung, Indonesia
(022) 82051 ext. 360
*Publishes "BERKALA ICUG" a monthly newsletter

Microcomputer Users' Group
Contact: Noble Bright
P.O. Box 1
Cape May, NJ 08204
(609) 884-2222
(609) 429-3838
Meets fourth Wednesday monthly

Northwest Association of Cromemco Users (NWACU)
Contact: Jim Illman
403 S. Brandon
Seattle, WA 98108
(206) 763-2099

North San Diego County Users' Group
Contact: Charles Mackey
P.O. Box 397
Fallbrook, CA 92028
(619) 728-6116
Located 30 mi. east of Oceanside

Assisting in the formation of Local Cromemco User Groups is one of the services performed by The IACU. We will be happy to provide you with a list of our members in your area, and recommend other contacts to help you organize and maintain a Cromemco computer users group. Just call or write I/O NEWS.

North Texas Cromemco Commercial Users' Group
Contact: Jerrell Johnson
1131 Winterwood
Lewisville, TX 75067
(214) 221-1437
Or call Rocky Hall
at (214) 398-1595
Meets first Wednesday bi-monthly

NY, NY Users' Group
Contact: Charles Perrella
45F Route 303
Valley Cottage, NY 10989
(914) 268-5137

SaCromemco Users
Contact: Alan Whitman
Box 244
Rancho Cordova, CA 95670
(916) 635-6070

Silicon Valley Cromemco Users
Contact: Allan O'Neill
(415) 969-3854 or Emily Ott (415) 854-5818
meeting place provided by:
MCM Enterprises
215 Hamilton Avenue
Palo Alto, CA 94301
Meets fourth Tuesday monthly

W.A. Cromemco Users' Group
Contact: Rae Canning
c/o The W.A. School of Computing
2/294, Rokeby Road
Subiaco, Western Australia 6008

West Germany Users' Group
Contact: Glynnis Long
Tesco GmbH
P.O. Box 10
8714 Weisentheid
West Germany
09383-1237
Total fluency in English & German

Wisconsin Cromemco Users' Group
Contact: Bob Ungemach
6249 West Browndeer Road
Browndeer, WI 53223
(414) 355-1451

COMMERCIAL MEMBER
Western United States

MULTI-MEDIA VIDEO INC.
3350 Scott Blvd., Bldg. 21
Santa Clara, CA 95051
Tel: (408) 727-1733
Tlx: 171-577 MMV USA

Multi-Media Video, (MVV), markets bilingual Arabic/English Cromemco systems and peripherals throughout the Middle East. Installations have been made in the government and banking sectors; a complete Arabic banking system was developed for the latter.

Key Personnel: A.B. Kader, President
Miguel Mora, Sales Manager
Jill Peterson, Marketing Manager

Major Market Area: Authorized dealers in Egypt, Saudi Arabia, and Pakistan.

SUDS

Continued from page 7

sion 2.41 and should be recompiled.

68000 Pascal Compiler (Model PAS-D-S/L)

Release 6 (version 2.42) of the 68000 Pascal programming language has few changes from the user's perspective, but provides substantial improvements in speed and efficiency.

68000 Fast C Compiler (Model FSTCCC-D-S/L)

Release 3 (version 2.41) of the 68000 Fast C programming language supports Cromemco's Maximizer co-processor board. The speed and efficiency of the Fast C compiler has been substantially improved, and the Symbolic Debugger has been slightly modified.

68000 Fast FORTRAN Compiler (Model FSTFOR-D-S/L)

Release 4 (version 2.41) of the 68000 Fast FORTRAN-77 programming language provides a substantial increase in speed and efficiency and a variety of new and revised features. Source code compiled by earlier versions of Fast FORTRAN-77 is incompatible with version 2.41, and should be recompiled.

68000 Fast Pascal Compiler (Model FSTPAS-D-S/L)

Release 4 (version 2.42) of the 68000 Fast Pascal programming language has few changes from the user's perspective, but provides substantial improvements in speed and efficiency.

68000 Fast BASIC-Plus Interpreter (Model FSTBAS-D-S/L)

Release 2 (version 2.40) of the 68000 Fast BASIC-Plus programming language supports Cromemco's Maximizer co-processor board. The speed and efficiency of the interpreter has been improved.

For Sale

Cromemco Color Graphics System including IMSAI S-100 Box, SDI-V, SDI-D, cabling, Necessary Software, RGB Monitor, 2 - 48KTPs, ZPU, 4FDC, Dual Persci Drives, and Televideo Monitor. Take it all, or make an offer on part of it. Call (916) 971-9610.

COMMERCIAL MEMBER

Western United States

EXCALIBUR COMPUTERS
4548 Auburn Blvd., Suite 191
Sacramento, CA 95841
(916 971-9610

Excalibur is a full service Cromemco Systems house. We provide Sales, Service, Warranty Repair and Support for Cromemco Products. We also provide full support for NEC spinwriters, Persci and Tandon Floppy Drives, and Televideo Terminals. We can provide training for CDOS, Cromix, Unix and a number of software programs. Custom programming applications can be written by us in 'C', sbasic, or any number of database query languages. We developed the **Excalibur Utilities** which include a number of programs designed to help construct menus and utilities which can be useful in many environments. We can upgrade 256KZs to 1024KZs, 256MSUs to 1024MSUs, 512MSUs to 2048MSUs, and DPUs to XPUs. We also sell the Hewlett-Packard Laserjet and standalone ST506 Drives at reduced prices. Call us for any training or support that you may need.

Key Personnel: Robert Brown, Sales and Marketing
Curt Johnson, Systems Engineer
Linda Wolfe, Customer Service

Inside Cromix

Continued from page 17

```
        cursor(24,1);
        printf("%s  %s",dat,timstr());  /* print today's date and time  */
        cursor(33,2);
        printf("CALENDAR FILE");
        cursor(33,3);
        printf("=============");

        zap(buff,MAXSIZE);
        k=0;            /* set line counter to zero     */
        tt=1;
        while((l=rdline(fn,buff,MAXSIZE)) != EOF)
        {                       /* read line including LF, ignoring CR   */
                buff[l-2]=buff[l-1]='\0';       /* change LF to NULL     */
                dd=rdat();
                if (dd <= tomorrow && tt <=tomorrow) /* ok to print item */
                {
                    cursor(1,k+4);
                    for (i=0;i<l-2;i++)         putchar(buff[i]);
                    putchar('\n');
                    if (k++ > 19)       /* if more than 19 lines  */
                    {
                        k=0;
                        cursor(34,23);
                        printf("More (Y/N)?  ");
                        setmode(STDOUT,MD_MODE1,0,ECHO); /* no echo   */
                        setmode(STDOUT,MD_MODE1,RAW,RAW);/* immediate */
                        do
                        {
                                while((c=getchar())==EOF);
                                c=toupper(c);
                        }
                        while(c != 'N' && c != 'Y');
                        setmode(STDOUT,MD_MODE1,ECHO,ECHO);
                        setmode(STDOUT,MD_MODE1,0,RAW);
                        if (c=='N')     break;
                                /* clear from line 3 to end of page  */
                        cursor(1,3);
                        printf(ctl,clreop);
                    }
                }
                if (dd) tt=dd; /*       takes care of no-date lines       */
                               /* will not print if preceding date not ok */
                zap(buff,MAXSIZE);
        }
        close(fn);
}
cursor(x,y)     /*   places cursor at col,row    home is 1,1    */
int x, y;
{
        printf(ctl,crlead);
        putchar( y + offset -1);
        putchar( x + offset -1);
}
long rdat()     /* returns value of date in line  10/21/85  format */
{
        int     i,j,k,l,m;
        long    tt;
        static char     dbuff[12];

        if(buff[0]=='\t' || buff[0]=='\n')      return(0);
        zap(dbuff,12);
        for(l=0;l<11;l++)  /* if tab or space get out   */
        {
                if (buff[l]=='\t' || buff[l]==' ')      dbuff[l]='\0';
                else                                    dbuff[l]=buff[l];
        }
        for(i=m=0;i<11;i++)
        {
            if(dbuff[i]=='/')   /* count /'s  */
            {
                dbuff[i]=' ';
                m++;
            }
        }
        if(m==2)        /*  if type 10/24/85    */
        {
            sscanf(dbuff,"%d %d %d",&i,&j,&k);
            tt=100*(100*k+i)+j;
            return(tt);
        }
        return(0);      /* all other cases              */
}
char    *timstr()                   /* returns stg with current time   */
{                                   /*  returns  10:03:24 AM   */
        int     hour,minute,second,i;
        char    time[4];
        static char     timout[12];

        zap(timout,12);
        gettime(time);
        hour  =time[0];
        minute=time[1];
        second=time[2];
        i=(hour > 11) ? 1 : 0;
        if (hour==0)    hour = 12;
        if (hour>12)    hour -=12;
        if(i)   sprintf(timout,"%2d:%02d:%02d PM",hour,minute,second);
        else    sprintf(timout,"%2d:%02d:%02d AM",hour,minute,second);
        return(timout);

```

Become a Member of The I.A.C.U. and have I/O News Sent Directly to You.

I.A.C.U. Application for Membership

Name: _____

Title: _____

Company: _____

Mailing Address: _____

_____

City: _____ State: _____

Country: _____ Zip / Postal Code: _____

Phone: _____ (Area code) _____ (number) _____ (extension)

Telex:: _____

Membership Rates in the United States:
( ) 1 yr. = $42.00 ( ) 2 yr. = $76.00
( ) 3 yr. = $99.00
Membership Rates in Canada and Mexico:
( ) 1 yr. = $49.00 ( ) 2 yr. = $90.00
( ) 3 yr. = $125.00
Membership Rates in all other countries:
( ) 1 yr. = $60.00 ( ) 2 yr. = $104.00
( ) 3 yr. = $149.00

☐ I've included an additional $8.95 for my three-ring reference binder (including first class postage in U.S., Canada & Mexico. International purchasers, please add $4.00 [U.S.] extra for postage.)

Please start my Membership in the International Association of Cromemco Users right away.

☐ I have enclosed my ( ) Check ( ) Money Order in the amount of $_____ (U.S. Dollars only please) for a ( ) year membership.
International applicants: add $10.00 (U.S.) if you prefer to pay through your local bank.

☐ Now Available in the U.S. and most other countries on your VISA or Mastercard
(U.S. Dollars drawn on U.S. banks only **please.** Please Type or Print Clearly)
( ) VISA / Mastercard _____ (Expiration Date)
_____ (Your Full VISA or Mastercard Number)
Date _____ Signature _____
_____ (Name exactly as it appears on card)

Attention CROMIX Users
☐ Include $25 + $4.95 handling for "CROMIX, A User's Guide" by Leigh Thomas. (Calif. add 6%)

Mail Your Membership Application to:
The International Association of Cromemco Users
P.O. Box 17658, Irvine, CA 92713 U.S.A.

We're Interested!

We're Really Interested...in what you have to say. Especially about how you use your system...the problems encountered and the solutions effected...unusual uses or environments...and any practical applications you would be willing to share with fellow members. These can be short notes for departments like 'bits & bytes...' and 'Tec Tips,' or full feature articles.

Contact Bill Jaenicke at I/O News for editorial guidelines or assistance. We're interested in unleashing your literary talents.

BACK ISSUES

of I/O News are available

CHANGE OF ADDRESS

Membership # _____
Type or Print
Your Name _____
New Address
_____
_____
New area code & phone No. ( ) _____
Sign Here _____
Date new address in effect _____

Send address changes to I/O News, P.O. Box 17658, Irvine, CA 92713

SYSTEMS ATLANTA, INC.

CROMIX SLAVE PROCESSORS
8MHZ Z80H 128K
Two users per slave
FAST — FAST — FAST — FAST — FAST
$1095.00 including software license

MAP100 MICRO ARRAY PROCESSOR
One 4MHZ AMD9511 Processor included
Card supports three processors
Single processor Z80 FORTRAN LIBRARY
Z80 SOURCE CODE NOW INCLUDED
MAKE Z80 FORTRAN 68000 FAST
SPECIAL PRICE — $395.00
Without AMD9511 — $195.00

NEW 32K STRUCTURED BASIC

CROMIX (SBASIC.BIN)	$295
PC/MS DOS	$295
CP/M	$195
UPDATE CROMIX VERSION	$ 95 *
UPDATE CDOS VERSION	$ 75 *

* Available to registered Cromemco Licensees only.

CP/M for Cromemco S100 Systems **$150**
FAST, interrupt-driven, 58K TPA, CDOS format-compatible
High speed diskette copy utility included

FAST BACKUP for CROMIX Hard Disks **$ 95**
Backup to floppy disk at one megabyte per minute, 8″ or 5.25″

FAST diskette copy programs **$ 95**
Copy both 8″ and 5.25″ CDOS, CROMIX and UNIX formats

Hard disk subsystems include STDC and custom cables.
Choice of attractive cabinet styles including rack mount and CS1

	Single	Dual
25 meg. med speed	$2595	$4195
36 meg. fast	3695	6395
52 meg. faster	4995	8995
85 meg. faster	5995	10995
140 meg. faster	6995	12995

Deduct $995 if STDC and custom cables are not desired.
One year warranty, extended warranty available, fast service.
Overnight replacement of defective drives covered by warranty.

Z80 cartridge tape backup utility **$95**
(Included with purchase of CTD tape subsystem at $1295)

Reliable, 16FDC-compatible 8″ floppy disks replace Persci 299's.

System Three's refurbished by exchange. No need to be down while work is being done.

Large selection of old and used Cromemco and compatible equipment at very low prices. Let us know what you want to buy and make an offer.

CALL OR WRITE FOR INFORMATION ON ALL PRODUCTS AND SERVICES.

SYSTEMS ATLANTA, INC., P.O. BOX 99, LEBANON, GA 30146 (404) 928-0240

CROMEMCO COMPUTERS: DESIGNED TO MAKE UNIX SYSTEM V EVEN BETTER...

UNIX System V, the new standard in multi-user microcomputer operating systems, gives you high performance features along with the portability and flexibility of a standard.

Cromemco computers can make UNIX System V even better. Because our systems are designed with UNIX in mind. First of all, we offer UNIX System V with Berkeley enhancements. Then, our hardware uses advanced features like 64K of on-board cache memory and our high speed STDC controller to speed up disk operations – very important with UNIX.

More capability and expandability

We have a high-speed, 68000-based CPU that runs at 10 MHz, coupled with a memory manager that uses demand-paging and scatter loading to work *with* UNIX, not for it.

We provide room for expanding RAM to 16 megabytes – with error detection and correction – for running even the most sophisticated and advanced microcomputer programs. And the power to accommodate up to 16 users – all with plenty of memory.

But we give you even more.

A complete solution

We give you a choice in systems: the System 100 series, expandable up to 4 megabytes of RAM, and the System 300 series, expandable to 16 megabytes. A high speed 50 megabyte hard disk drive is standard on the systems. And you can expand the hard disk capacity up to 1200 megabytes using standard SMD drives. You can add floating point processing. High resolution graphics. Video digitizing and imaging. Communications through standard protocols. Mainframe interface.

And software support is here to meet your needs. We offer major programming languages, database management systems, communications software, including SNA architecture, X.25 protocol, and Ethernet; even a program to interface to an IBM PC if you need to. And, of course, access to the broad range of standard UNIX applications programs that is growing dramatically every day.

Easy to use.

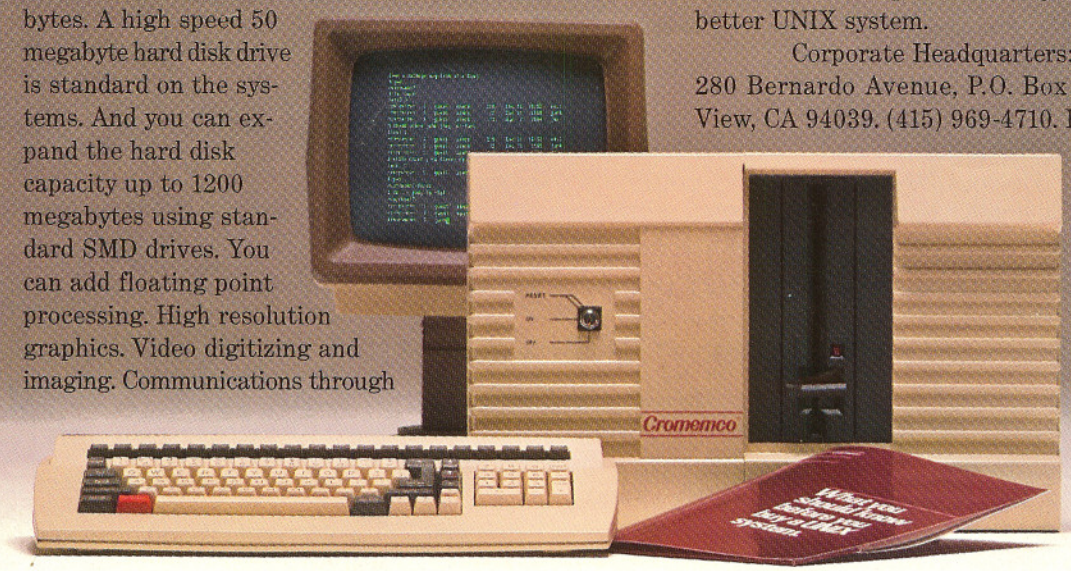
We also make our systems easier to use, because we install the operating system before we ship your computer. No complicated installation procedures. And the Berkeley enhancements give you the standard UNIX System V operating system, but with the added convenience of these widely acclaimed improvements.

Cromemco's System 100 and System 300 computers: designed to be the highest performance UNIX systems available anywhere.

Just call or visit one of our UNIX System V Official System Centers to see for yourself. They'll also give you a copy of our new publication, "What you should know before you buy a UNIX system." Or contact us directly.

We'll be glad to show you how to get a better UNIX system.

Corporate Headquarters: Cromemco, Inc., 280 Bernardo Avenue, P.O. Box 7400, Mountain View, CA 94039. (415) 969-4710. In Europe: Cromemco GmbH, 6236 Eschborn 1, Frankfurter Str. 33-35, P.O. 5267, Frankfurt Main, Germany.



UNIX is a trademark of Bell Laboratories.
IBM is a trademark of International Business Machines Corp.

Cromemco®